DEFENSE INFORMATION SYSTEMS AGENCY

JOINT INTEROPERABILITY AND ENGINEERING ORGANIZATION

CENTER FOR INFORMATION MANAGEMENT

SOFTWARE SYSTEMS ENGINEERING DIRECTORATE

Ada Dual-Use Summary

Ada Dual-Use Workshop
October 19-20, 1993

Ada Dual-Use Committee Briefing
November 8, 1993

DEFENSE INFORMATION SYSTEMS AGENCY

JOINT INTEROPERABILITY AND ENGINEERING ORGANIZATION

CENTER FOR INFORMATION MANAGEMENT

SOFTWARE SYSTEMS ENGINEERING DIRECTORATE

Ada Dual-Use Summary

Ada Dual-Use Workshop
October 19-20, 1993

Ada Dual-Use Committee Briefing
November 8, 1993

REPORT DOCUMENTATION PAGE

Form Approved

OPM No.

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources gathering and maintaining the data needed, and reviewing the collection of information. Send comments regading this burden, to Washington Headquarters Service, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Information and Regulatory Affairs, Office of Management and Budget, Washington, DC 20503.

1. AGENCY USE **(Leave**

2. REPORT: 29 NOV 93

3. REPORT TYPE AND DATES: Proceedings for October 19-20, 1993 Workshop

4. TITLE AND: Ada Dual-Use Summary,
subtitle: Ada Dual-Use Workshop, October 19-20, 1993
Ada Dual-Use Committee Briefing November 8, 1993

5. FUNDING

6. AUTHORS:

Various

7. PERFORMING ORGANIZATION NAME (S) AND. Various

8. PERFORMING ORGANIZATION

9. SPONSORING/MONITORING AGENCY NAME(S) AND: Defense Information Systems Agency (DISA), Joint Interoperablility and Engineering Organization (JIEO), Center for Information Management (CIM), Software System Engineering Directorate (TXE)

10. SPONSORING/MONITORING AGENCY

11. SUPPLEMENTARY

12a. DISTRIBUTION/AVAILABILITY: Approved for public release; distribution unlimited

12b. DRISTRIBUTION

13. ***(Maximum 200***

Proceedings

14. SUBJECT: Ada Programming Language, Dual-Use

15. NUMBER OF

16. PRICE

17 SECURITY CLASSIFICATION	**18. SECURITY**	**19. SECURITY CLASSIFICATION**	**20. LIMITATION OF**
UNCLASSIFIED	**UNCLASSIFIED**	**UNCLASSIFIED**	**UNCLASSIFIED**

NSN

DEFENSE INFORMATION SYSTEMS AGENCY
JOINT INTEROPERABILITY AND ENGINEERING ORGANIZATION
11440 ISAAC NEWTON SQUARE, NORTH
RESTON, VIRGINIA 22090-5006

IN REPLY REFER TO: TXE

29 November 1993

MEMORANDUM FOR DISTRIBUTION

SUBJECT: Ada Dual-Use Workshop Proceedings

1. On behalf of the Defense Information Systems Agency (DISA), I would like to express my sincere gracitude to you for your participation during the Ada Dual-Use Workshop that was held on October 19-20, 1993, at the Sheraton Premiere-Tysons Corner in Vienna, Virginia.

2. The suggestions, information and recommendations that you provided to the Department of Defense (DoD) will promote the dual-use of Ada throughout the software industry for many years to come.

3. Enclosed are the proceedings from the workshop, as well as the recommendations that were provided to the Ada Dual-Use Committee on November 8, 1993 by the respective chairs of the four panel sessions regarding the government, academia, vendor and industry areas.

4. You may rest assured that we at DISA are committed to the Ada programming language and the dual-use initiative. Again, thank you for making this effort an on-going success. If you have any further questions, please call my point of contact Mr. Jerry Russomano at (703) 285-6589.

BRUCE J. BOHN
Brigadier General, USAF
Director

2 Enclosures:
1 Ada Dual-Use Committee Meeting Recap
2 Press Release

ADA DUAL-USE COMMITTEE MEETING RECAP

The Ada Dual-Use Committee met on November 9, 1993 to review the findings, conclusion and recommendations of the DoD sponsored Ada Dual-Use Workshop that was held last month in Vienna, Virginia. The meeting, which was chaired by the Honorable Emmett Paige, Jr., Assistant Secretary of Defense for Command, Control, Communications and Intelligence (ASD/C3I), consisted of several senior level executives and flag officer representatives from within the Department of Defense. The Ada Dual-Use Committee members included LTG Alonzo E. Short (DISA), Lt Gen Albert J. Edmonds (J6), LTG Kind (USA), Lt Gen Carl O'Berry (USAF), Mr. Lloyd Mosemann (SAF/AQK), RADM John G. Hekman (USN/NISMC), Brig Gen Bruce J. Bohn (JIEO), and Mr. Ronald Elliott (HQ/USMC), Ms. Cynthia Kendall (OASD (IM)), and Dr. Howard Frank (ARPA).

Presentations were made to the Committee members by each of the following Panel Chairs; academia: (Dr. Charles McKay), government: (Dr. Kurt Fisher), industry: (Dr. Terry Straeter) and Vendor: (Mr. Bill Carlson). A summarization of recommendations and an action plan were then presented by Mr. Jerry Russomano and Mr. Donald Reifer on behalf of the Defense Information Systems Agency, Joint Interoperability and Engineering Organization, Center for Information Management (DISA/JIEO/CIM).

Upon conclusion of the meeting, Mr. Paige strongly endorsed the Department of Defense's commitment to Ada. DISA will proceed to implement the following action items:

1. A detailed Program Plan for the DoD Ada Dual-Use Initiative should be developed to itemize and prioritize investments in each of the recommended strategy areas (Office of Primary Responsibility: DISA/JIEO and the Ada Joint Program Office (AJPO).

2. Migration systems should be pursued as the major target of opportunity for the DoD Ada Dual-Use Initiative (Office of Primary Responsibility: DISA and the Services).

3. Current marketing efforts should be expanded to increase Ada awareness and appeal to commercial firms (Office of Primary Responsibility: DISA/JIEO and the AJPO).

4. Investments in bidings (Ada interfaces to operating systems, tools/environments, frameworks, commercial off-the-shelf packages, etc.) tools, education, and training should be pursued under the Ada Technology Insertion Program (Office of Primary Responsibility: DISA/JIEO and the AJPO).

5. Strategic partnerships with academia, industry and other government agencies should be used whenever possible to achieve Dual-use goals. Such partnerships require all participants to invest equally in developments (Office of Primary Responsibility: DISA/JIEO and the AJPO).

6. Reward structures should be revamped to provide positive incentives to both contractors and government personnel who demonstrate excellence in Ada (Office of Primary Responsibility: DISA/JIEO and the AJPO).

7. DoDD 3405.1 should be rewritten to clarify misinterpretations and provide consistent guidance (Office of Primary Responsibility: DASD (IM)).

8. Reuse libraries need to be openly identified to distribute Ada Reusable Software Components (RSCs) to the software engineering community (Office of Primary Responsibility: Software Reuse initiative, DISA/JIEO and the AJPO).

9. AdaSAGE source code should be disseminated to the Ada software community. (Office of Primary Responsibility: DISA/JIEO and the AJPO).

Proceedings of the DoD sponsored Ada Dual-Use Workshop will be distributed during December 1993. Additional copies and general information may be obtained from the Ada Information Clearinghouse (Phone: 1-800-232-4211).

An Ada Dual-Use Initiative Workshop is being planned for the fall of 1994. This Workshop will discuss accomplishments during FY94 and detail plans for the coming year.

DEFENSE INFORMATION SYSTEMS AGENCY
JOINT INTEROPERABILITY AND ENGINEERING ORGANIZATION
CENTER FOR INFORMATION MANAGEMENT
701 S. COURT HOUSE ROAD
ARLINGTON, VIRGINIA 22204-2199

IN REPLY
REFER TO: TXE

PRESS RELEASE

In cooperation with academia, industry and other government agencies, the Department of Defense has been exploring ways to increase the commercial use of Ada. An Ada Dual-Use Committee meeting chaired by the Honorable Emmett Paige, Jr., Assistant Secretary of Defense for Command, Control, Communications and Intelligence (ASD/C3I) was held on November 8, 1993 to review the findings, conclusions and recommendations of the Ada Dual-Use Workshop that met during October 19-20, 1993 in Vienna, Virginia.

Upon conclusion of the meeting, Mr. Paige endorsed the continued use of Ada and directed the Defense Information Systems Agency, Joint Interoperability and Engineering Organization (DISA/JIEO) to develop an Ada Dual-Use Initiative Program Plan aimed at executing the actions recommended by the Workshop panel chairs beginning this year. These actions are aimed at promoting Ada more effectively, providing support and incentives, and enforcing the Ada programming language mandate.

The proceedings of the DoD sponsored Ada Dual-Use Workshop will become available in early December. The Program Plan is expected to be released during mid-January 1994.

ADA DUAL-USE
COMMITTEE BRIEFING

JERRY RUSSOMANO
DIRECTOR, SOFTWARE SYSTEMS ENGINEERING
CENTER FOR INFORMATION MANAGEMENT

OUTLINE

- Academia Panel Summary
- Government Panel Summary
- Industry Panel Summary
- Vendor Panel Summary
- Workshop Summary
- Committee Discussion (Groupware)
- Committee Recommendations (Groupware)

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 2 - Outline

Approximately 175 individuals participated in the Ada Dual-Use Workshop, held on October 19-20, 1993. As you know, DISA has been assigned the responsibility to evaluate current Ada initiatives and align the strategies of the commercial and government sectors. The main objective of the Workshop was to identify ways to increase the use of Ada programming language and related technology in both defense and commercial applications. To accomplish this goal, the workshop was organized into the following four panels; academia, government, industry and vendors. Each panel deliberated to determine ways to make Ada more attractive to both DoD and commercial users. The Ada Community views the Workshop as a success. The community's testimony is reflected in the conclusions and recommendations which will be summarized and presented by the panel chairs during this briefing. After the panel chairs have completed their briefings, DISA's commercialization strategies as derived from the Ada Dual-Use Workshop will be presented. The positive actions identified for the Ada commercialization effort are a direct result of the Ada Community's participation with the Workshop. Comments and recommendations from the Ada Dual-Use Committee will be solicited using a computer-aided facilitated discussion tool.

DEFENSE INFORMATION SYSTEMS AGENCY

Academia Panel Summary

Dr. Charles McKay
University of Houston - Clear Lake

DEFENSE INFORMATION SYSTEMS AGENCY

Academia Panel

Issue Areas

- Education and Research
- Software Engineering Principles
- Communication

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 4 - Issue Areas

Listed are three issue areas as concluded by the Academia Panel; education and research, software engineering principles, and communication.

Education and Research

- Provide low cost/free environments, tools and other resources supporting instruction and research in and with Ada 9X to educators and students as soon as possible
- Libraries of reusable components useful for instruction and research should be made available at little or no cost to educators
- An Ada-based, software engineering research agenda for at least the next five years should be established and publicized
- Fund instruction, research and service that helps place Software Engineering and Ada 9X in "front of the wave"

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 5 - EDUCATION AND RESEARCH

Such support should be consistent with the software engineering goals and objectives of Ada 9X and rewarding for the initial and the more experienced user. At the entry level, the visual interfaces and other support for student learning must be at least as good as what is now provided in competing languages and environments such as C++ and Turbo Pascal (trademarks).

Such components should span the software engineering life cycle and include bindings, interfaces and implementation components.

Such an agenda should include the kind of large-scale, basic research issues now addressed by ARPA initiatives. At the other end of the spectrum, the agenda should also include opportunities for smaller-scale, applied research projects that might be addressed by new faculty members in smaller institutions.

Strengths of Ada 9X, such as reliability and maintainability, are not as attractive to potential students as perceived applicability to such current hot topics as virtual reality. However, virtual reality is only one of the current hot topics that could benefit from Ada 9X. Sponsorship of Ada 9X based research in such subjects along with sponsorship of student team competitions can raise the visibility and desirability of including Ada 9X and Software Engineering in academia.

Software Engineering Principles

- DoD should sponsor an initiative to effect the curriculum improvement that covers and integrates issues across the software engineering life cycle
- Restore/initiate consistency in DoD's treatment of Ada and Software Engineering
- Coordinate a long term, cooperative program among government, industry, vendors and academia to advance the goals and objectives for Software Engineering and Ada 9X

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 6 - SOFTWARE ENGINEERING PRINCIPLES

The initiative should include the use of Ada 9X in a variety of domains including, but not limited to, mission critical embedded systems, C3I and MIS. Such an initiative should build upon other efforts such as the ARPA BAA and the AJPO TIP. In addition to ASEET, the SEI should serve as a focal point for this work.

Say what you mean. Mean what you say. Budget and act accordingly. The supporters of Ada and Software Engineering within academia have paid a very high price for the vacillations and inconsistent signals of DoD. A mandate that is ignored without consequence, that is unsupported by any major, Ada-based research initiatives in academia, and that is not generally perceived to relate to anything other than the issues of DoD's programming language preferences undermines the best interests of all stakeholders and strongly argues against the inclusion of Ada in academia.

Sponsorship of thesis students, faculty and student researchers, cooperative research and instruction, internships and exchange programs are some examples of such coordination.

Communication

- Expand an existing forum, ASEET (Ada Software Engineering Education and Training), to serve the needs for timely information and technology exchange among all educators that are stakeholders in Ada 9X
- NSF-style faculty development workshops should be made available to facilitate the expansion and integration of Software Engineering and Ada 9X instruction and research in academia
- An outreach program should be mounted in conjunction with implementing the above recommendations so that the necessary public relations, marketing and other technology transfer activities support the timely insertion of Software Engineering and Ada 9X into education and training

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 7 - COMMUNICATION

Membership should include, but not be limited to, educators from: military academies, universities, colleges, technical institutes, community colleges, high schools, commercial training firms and Software Engineering/Ada instructors within an organization.

This means travel, per diem and a small stipend should be available for educators who are interested in participating in such workshops.

The Ada IC should play an important role in this dissemination.

Government Panel Summary

Dr. Kurt Fischer
OASD (C3I), ODASD (IM)

DEFENSE INFORMATION SYSTEMS AGENCY

Government Panel

Ada Issue Areas

- Policy and Enforcement
- Publicity
- Bindings
- Acquisition
- Tools
- Education and Training

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 9 - Issue Areas

Listed are six issue areas as concluded by the Government Panel; policy and enforcement, publicity, bindings, acquisition, tools and education and training.

Policy and Enforcement

- Clarify Ada Policy
- Reiterate policy to components
- Encourage subsets and supersets
- Review Ada compliance at oversight reviews
- Audit programs for compliance
- Establish a public list of waivers

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 10 - POLICY AND ENFORCEMENT

Clarify and Promulgate Ada Policy

- OASD (C3I) reword Ada policy to clarify:
  - Use of Ada in research and development
    - When waivers are required
- ASD (C3I) send letter to components reiterating Ada policy/rationale and need for strong compliance.
- Encourage subsets and supersets. especially for research and development and prototyping; continue use of only validated Ada compilers for production.

Strengthen Ada Policy Enforcement

- Ada Compliance to be a special interest item and exit gate at all MAISRC and DAB reviews.
- ASD (C3I) direct components to make Ada compliance a special interest item for all component oversight reviews.
- ASD (C3I) direct audit of all C3I programs and initiate transition to Ada for non-compliant programs or pursue waivers.
- Director, DISA audit DISA programs and initiate transition planning to Ada for non-compliant programs or pursue waivers.
- Establish and make public the list of Ada waivers.

DEFENSE INFORMATION SYSTEMS AGENCY

Publicity

- Publicity and marketing are as important to Ada's success as the language and tools
- Ada success stories must be better packaged and communicated
- Ada success in DoD must be better recognized and rewarded
- Ada must be part of American education at all levels

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 11 - PUBLICITY

Ada success in DoD must be better recognized and rewarded

- Hold executive breakfast meetings around the country; have top draw speaker describe Ada 9X benefits and Ada '83 successes.
- OASD/C3I recognize Program Managers and PEOs of Ada-intensive projects with annual awards banquet.
- Sponsor prize for "Best Commercial Application Using Ada" and present at annual banquet. Target audience to be CIOs and VPs of Product Development.

Ada must be part of American education at all levels

- Fund and actively promote the Free Software Foundation Ada compiler with 9X enhancements, especially promote to colleges and universities.
- Donate old versions of Ada tools to colleges and universities.
- Encourage/mandate teaching of Ada at all DoD universities and academies.
- Distribute government sponsored tools and tool prototypes widely, especially colleges and universities.

Publicity and marketing are as important to Ada's success as the language and tools

- Hire a publicist and develop a publicity plan.
- Develop (buy) a comprehensive mailing list.
- Form an Ada Publicity Alliance with Ada vendor publicists; meet annually to recommend DoD publicity actions.
- New AJPO Director must have excellent marketing skills and be able to relate to entire software community.

Ada success stories must be better packaged and communicated

- Collect metrics and report findings that demonstrate Ada benefits.
- Seek and publish Ada success stories in commercial publications.
- OASD/C3I grant spot awards ($500) for Ada related articles in non-Ada press.
- Host Ada booth at commercial trade shows (e.g., COMDEX).
- Publish CrossTalk journal
- Produce videos describing Ada and its benefits for corporate executives, technical personnel, and students (all levels).
- Cultivate non-Government technology press and better publicize Ada events and milestones.
- Publish "The Impact of Ada and DoD", now underway by Joint Logistics Commanders.

Bindings

- DoD publish binding availability
- DoD develop and standardize missing binding specs
- DoD hold Ada Vendor Conference to encourage binding product development
- DoD develop Ada architecture framework
- DoD develop Ada strategy to coexist with legacy systems

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 12 - BINDINGS

The DoD needs to review the availability, quality, and source of binding specifications and implementations.

The DoD should initiate a focused program to develop and standardize binding specifications (in cooperation with vendors and users) to popular interfaces such as POSIX, SQL, DOS, Microsoft Windows, X, PCTE, CORBA, etc.

The DoD should encourage vendors to develop binding products by establishing an Ada Vendors Conference (e.g., at George Mason University) where the DoD could:

- Reiterate its Ada policy
- Describe the availability of Ada bindings
- Describe its need for bindings
- Describe and receive feedback on its plan to communicate binding availability from DSRS

The DoD needs to facilitate Ada use by developing and promulgating an architectural framework for Ada.

The DoD should develop and promulgate a strategy for Ada software to coexist with legacy software and data

Acquisition

- Reform acquisition process to include lifecycle multi-year strategic planning for all software systems
- Work with acquisition community to define acquisition-related recommendations

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 13 - ACQUISITION

Reform acquisition process to include lifecycle, multi-year strategic planning for all software systems.

- Development and transition of required 6.1 and 6.2 technology
- Development and management of mission/business processes, domain architectures, class libraries, software components, and specialized tools.
- Maintenance and support community coordination.

DISA to work with DASD(A) and USD(A) acquisition community to define acquisition-related recommendations.

Tools

- DoD encourage Ada technology advancement
  - Use SBIR contracts
  - Enhance and distribute AdaSAGE
  - Subsidize development of Visual Ada
- Open DSRS to all U.S. organizations and publicize and share GOTS tools and libraries
- Include Ada tools in umbrella hardware/software procurements
- Allow Ada 9X compilers in validation for use in non-critical production systems

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 14 - TOOLS

Ada development environments are lagging behind "C" and other language environments. The PC environment is in dire need of high-quality, low-cost Ada compilers and tools. The DoD needs to use its buying power to require better support environments.

By opening up reuse libraries to U.S. organizations, we can provide quality software to the Ada world.

High discount procurement vehicles are frequently unavailable to some program managers. DoD should include Ada tools in umbrella hardware and software procurements.

The Ada community cannot afford to wait six months for tools to support Ada 9X. Otherwise, Ada 9X will fail. Also, the DoD should look at allowing non-validated Ada 9X compilers in the market until validated compilers exist.

Education and Training

- Promote Ada within education community
  - Publicize education benefits of Ada with reports and videos
  - Make public domain/GOTS tools available through DSRS
  - Support Internet education discussions
- Mandate Ada education at service academies and other DoD schools
- Promote Ada 9X textbook development
- Provide university grants to do Ada research
- Increase PEO/PM Ada benefits training

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 15 - EDUCATION AND TRAINING

Promote Ada within the education community

- Publicize the educational benefits of Ada at all education levels. Develop and publish a report highlighting Ada benefits for each level of education (primary, secondary, college).
- Open up the DSRS to all U.S. organizations, especially educational institutions and make public domain and GOTS tools and libraries available.
- Provide the education community relevant information (training, tools, textbooks, research) through Internet USENET discussions.

Mandate Ada education at service academies

- OASD (C3I) review and recommend Ada policy position.
- ASD (C3I) visit academies and discuss Ada with Provost.
- ASD (C3I) request USD(A) to require relevant Ada management/technical training forming core Defense Acquisition.

Promote Ada 9X textbook development

- Break GPO log jam

Provide grants to universities to do Ada research.

Increase PEO/PM Ada education and training to assure through teaching of Ada benefits.

DEFENSE INFORMATION SYSTEMS AGENCY

Industry Panel Summary

Dr. Terry Straeter
GDE Systems, Incorporated

DEFENSE INFORMATION SYSTEMS AGENCY

Industry Panel

Issue Areas

- Enforcement of DoD Mandate
- Business case for Ada
- Bindings and Tools
- Marketing and Promotion
- Education and Training
- Reuse Libraries
- Incentives

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 17 - Issue Areas

Seven issue areas as concluded by the Industry Panel; enforcement of DoD mandate, business case for Ada, bindings and tools, marketing and promotion, education and training, reuse libraries, and incentives.

Enforcement of DoD Mandate

- The mandate is not working
- To strengthen it, the following actions are needed:
  - Eliminate the waiver process
  - Pull R&D, FFRDCs and other non-compliant organizations (NSA, etc.) in line
  - Provide real incentives to contractors with proven Ada capabilities (award preferences, increased fee, etc.)
- DoD needs to be consistent in its interpretations and speak with one voice

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 18 - Enforcement of DoD Mandate

The Industry Panel Session findings indicated that the DoD mandate is not working. The DoD should strengthen the Ada mandate by either eliminating or enforcing the waiver process and adhering to public law.

Reinforce the DoD's language policy to include high integrity, research and development and information systems where life cycle cost issues are paramount. All R&D, FFRDC's, and other non-compliant organizations (NSA, etc.) should be brought in-line with the Ada directive.

Provide real incentives for those firms that have proven Ada capabilities on legacy systems. This could include award preferences, or increased fee Awards could be given to DoD Program Managers and firms who demonstrate Ada competence during contract performance.

DoD needs to be consistent in its interpretations and speak with one voice. The panel findings indicated that the DoD's application of the Ada mandate is not consistently applied across government sectors and is often ignored.

Business Case for Ada

- DoD needs a convincing business case to justify the use of Ada to middle managers
- To be successful, Ada must be viewed as a success
- DoD needs to:
  - Capture success stories
  - Provide "hard" data
  - Create the image of success
- Existing efforts to develop a business case need to be accelerated

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 19 - Business Case for Ada

To be successful, Ada must be viewed as a success. The DoD needs to place a business focus on Ada plans and strategies. The DoD should accelerate and cooperate with existing efforts (SIGAda, Software Alliance, etc.) to substantiate Ada's excellence. A solid business case and briefing should be developed for Ada with global statistics that represent "hard" (quantitative) data on comparative performance (cost, schedule, quality, etc.) of quality issues throughout the life cycle. The DoD should create the image of success for Ada. Success stories should be captured and distributed. Discussions should focus on why Ada was selected and what benefits results from its use.

Bindings and Tools

- DoD needs to increase the attractiveness of its Ada products by providing bindings and interfaces with:
  - Window's toolkits
  - C/C++ libraries
- DoD should change its strategy from confrontation to cooperation with the C/C++ community
- Make Ada attractive to its competitors and you'll increase your chances of success
- Take advantage of the second "window of opportunity" while the iron is hot

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 20 - Bindings and Tools

There is a need for more bindings and COTS tools. The DoD needs to increase the attractiveness of its Ada products by providing bindings and interfaces with Window's toolkits and C/C++ libraries. The DoD should change its strategy from confrontation to cooperation with the C/C++ community. The DoD should establish partnerships with Ada vendors to jointly develop high demand items on a cost-sharing basis (Government could take limited rights; contractor could get marketing rights). The DoD's strategy should look toward coexistence, not toward exclusive use of Ada. If you make Ada more attractive to its competitors, you will increase your chances of success. The DoD needs to take advantage of the second "window of opportunity" while the opportunity exists.

Marketing/Promotion

- DoD has done a terrible job marketing/promoting Ada
- DoD needs to move Ada out of the hands of the technologists and into the hands of business people
- DoD revitalization efforts should be market-driven
- Proven marketing tactics should be used to pinpoint, solidify and exploit market opportunities
- A marketing/promotion plan should be a first order of business

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 21 - Marketing/Promotion

Ada has not been well marketed. The DoD should put more of a business focus on Ada plans and strategies. A marketing strategy for the commercialization of Ada needs to be defined. The focus of the strategy should be to move Ada from a technology-driven strategy to one that is market-driven. Proven marketing tactics should be used to pinpoint, solidify and exploit market opportunities. A marketing/promotion plan should be a first order of business and should include a plan for improving the outreach of Ada. This could be accomplished by establishing a marketing objective for the AJPO that includes interacting with professional societies, FFRDCs, trade associations, other government agencies, universities, and industry.

Education and Training

- Ada should be emphasized along with software engineering as a core competency in the DoD
- DoD schools and institutions should be directed to tow the line
- Incentives (grants, tools, etc.) should be provided to motivate universities to teach Ada
- Preference in hiring for Ada trained specialists should be provided
- Availability of Ada 9X courseware should be immediate priority

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 22 - Education and Training

Ada should be emphasized along with software engineering as a core competency in the DoD. The DoD should place a greater emphasis on training government personnel. This could be accomplished by providing videos and other multi-media based educational software for self-paced management and programmer education. DoD schools and institutions should be directed to support Ada. Incentives such as grants, tools, etc. should be provided to motivate universities to teach Ada. In addition, partnerships should be formed to stimulate educational use of Ada. The DoD should help support an Ada teaching environment by providing low-cost compilers for Ada-based software engineering training. The DoD should provide preference in hiring for Ada trained specialists. Availability of Ada 9X courseware should be immediate priority.

Reuse Libraries

- Ada should be central to the DoD's reuse thrust
- Libraries should be opened to permit easy access to DoD Ada reusable software components
- Reuse libraries should be used as means to distribute high demand items
- Synergy can be achieved by pushing forward with an Ada-based strategy

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 23 - Reuse Libraries

The community as a whole needs to provide reusable Ada libraries for all environments, domains, and platforms. Ada should be central to the DoD's reuse thrust. The DoD's software reuse libraries should be opened to permit easy access to Ada reusable software components. The DoD should recognize and use reuse repositories as a means to distribute high demand items. Synergy with commercial markets and other government agencies can be achieved by pushing forward with an Ada-based reuse strategy.

Incentives

- Both push and pull are needed to make Ada attractive within DoD and to commercial industry
- Innovation should be encouraged and the phrase "I can't do that" abolished
- Partnerships should be formed and financial rewards offered
- Government personnel should be rewarded/penalized accordingly

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 24 - Incentives

The DoD should provide incentives that will motivate investors to renew their Ada initiatives, and should stimulate industry to develop and recognize core competency in Ada. Both "push and pull" are needed to make Ada more attractive within the DoD and the commercial industry. The DoD should encourage innovation and work to abolish the phrase "I can't do that." Strategic partnerships should be formed with other government agencies, universities and colleges, professional societies and trade associations, and dual-use programs (NIST, ARPA, NSF, etc.). The DoD should offer financial rewards for Ada use. Government personnel should be rewarded or penalized accordingly.

Summary

- A broad-based initiative should be mounted and headed by a "leader in the field"
- Funds should be allocated and the program should be managed as a major element of DoD's dual-use strategy
    - Need $100M over 5 years
    - Need ARPA, NSF and NIST support
    - Need to put up or be quiet
- Hope you will make the commitment

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 25 - Summary

The DoD should mount a broad-based Ada commercialization initiative headed by a "leader in the field". Funds for the Ada program should be allocated and the program should be managed as a major element of the DoD's dual-use strategy. In order to plan and implement an effective Ada commercialization strategy, the DoD needs do the following:

- Allocate funds of $100M over 5 years
- Enlist ARPA, NSF, and NIST support
- Commit the funds and appropriate resources or withdraw Ada mandate

The DoD needs to show both the commercial and government sectors that it is serious about its commitment to Ada.

DEFENSE INFORMATION SYSTEMS AGENCY

Vendor Panel Summary

Mr. Bill Carlson
Intermetrics, Incorporated

VENDOR PANEL

Issue Areas

- DoD commitment to use Ada
- Marketing
- Business environment conducive to private investment
- Role of Ada 9X

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 27 - Issue Areas

Listed are four issue areas as concluded by the Vendor panel; DoD commitment to use Ada, marketing, business environment conducive to private investment, and role of Ada 9X.

DoD Commitment

- Prepare and distribute a succinct statement of why DoD is committed to language standardization and to using Ada
- Demonstrate commitment through concrete actions
- DISA to lead use of Ada 9X for information systems applications - start reengineering at least one migration system into Ada 9X this year

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 28 - DOD COMMITMENT

The Ada vendor community unanimously rated DoD commitment to Ada and related success in the DoD software sector as the most important pre-conditions for success in the commercial sector. The DoD's inconsistent posture on Ada and its failure to follow through on its Ada mandate are serious impediments to Ada dual use. Some of the cited incidents, which need eliminating, are:

- Ada not taught at the service academies
- Military and civilian personnel ignoring policy, law, and directives with impunity
- Lack of 6.1 and 6.2 funds for research supporting Ada
- The selection of a COBOL finance system over Ada alternatives (STANFINS)

The DoD must make a clear consistent statement of its commitment to Ada and demonstrate that commitment through concrete actions. By explaining why the DoD is committed to Ada, the Ada community, both inside and outside of the DoD, will have a clear understanding of the Ada's role in the DoD. The DoD needs to demonstrate this commitment through:

- Public scrutiny for non-use.
- Penalties for non-use.
- Incentives for Ada use.
- Numerous announcements of specific systems which are being converted or implemented in Ada.

There is no better way to show commitment, than by investing in your own plan -- DISA should lead the use of Ada 9X by converting one migration system to 9X this year. We suggest a budget of $2M - $3M this year, and recommend that the objectives of this initial migration effort (i.e., size and complexity of system) be established appropriate to that level of effort.

DEFENSE INFORMATION SYSTEMS AGENCY

Marketing

Enhance Awareness of Ada's Benefits

- Focus DoD's participation in Ada marketing on DoD's strengths - publicizing DoD's successful uses of Ada in a forceful and dramatic way
- Vendors and Ada Software Alliance take the lead in promoting Ada outside DoD

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 29 - MARKETING

Ada is beginning to receive the attention and respect it deserves in the non-Ada community, but vendors would like to see a much higher level of favorable public awareness. Vendors would like to see a wider dissemination of Ada success stories from DoD applications. Vendors want to work with the DoD to present success stories in a way which will be effective to the non-DoD community. The DoD needs to increase its marketing of Ada by:

- Having senior decision-makers highlight Ada's benefits and DoD's rationale for using it.
- Encouraging DoD personnel and contractors to publish technical articles and give presentations at technical conferences which provide the facts and benefits of using Ada.
- Making it clear that it is appropriate for DoD personnel to be "references" for Ada products and services which have been used effectively on DoD projects.
- Developing, in partnership with commercial companies, effective presentations of the business case for Ada.
- Explaining DoD's rationale for committing to Ada, to civilian government personnel, especially targeting personnel involved in joint programs with the DoD.
- Vendors want to work with the DoD to help market Ada. The DoD needs to make it clear that DoD personnel are encouraged to work with vendors in enhancing the public awareness of Ada benefits, business cases, and success stories.

Business Environment Conducive to Private Investment

- DoD focus on its need to have software written in Ada
- DoD to encourage/incentivize dual-use products
- Provide funding for Ada related research in universities

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 30 - BUSINESS ENVIRONMENT CONDUCIVE TO PRIVATE INVESTMENT

The Ada market is substantial -- the current market for Ada compilers and tools is about $300M per year and there are about 50,000 people using Ada, today! However, overly optimistic estimates of an extremely large marketplace during Ada's formative years may hinder efforts to attract venture capital funding to launch a dual-use initiative. In order to stimulate the dual-use of Ada, vendors want to establish a partnership with government with clearly defined roles. Vendors want DoD to focus its investment dollars on standards and customer applications which meet specific DoD needs, not on the development of software products which compete with industry.

The DoD should focus first and foremost on the need to have its own software written in Ada. The DoD should be rock-solid in its commitment to Ada and it should have a plan to implement all new systems in Ada and to transition legacy systems to Ada. The plan should be implemented and tracked, and progress should be periodically released.

If someone will be writing DoD software, DoD should make sure that they have an Ada compiler and tools, including CASE tools that support development and production of quality Ada software. DISA's broad software process assessment capability should be used to monitor Ada usage within DoD Central Design Activities and at DoD developer sites.

Role of Ada 9X

- Ada 9X is a chance to reinvent Ada's image
  - Safe reliable object oriented programming
  - Support multiprocessing and distributed systems
- Accelerate availability of high quality Ada 9X implementations with monetary incentives (investment sharing)
- Plan and implement early application of Ada 9X within DoD

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 31 - ROLE OF ADA 9X

Ada 9X extends Ada with features which provide object-oriented capabilities and improved consistency of real-time performance. DoD personnel and contractors should use Ada 9X as soon as there are proven quality implementations. Ada 83 is an excellent tool, and there is no reason to rush the transition to Ada 9X. Once high quality implementations of Ada 9X can be delivered, an excellent strategy for expanding commercial use of Ada is to position Ada 9X as a new language providing safe reliable object-oriented programming.

The DoD should invest at least $10M this year to buy Ada 9X compilers, tools, and bindings for future delivery. This DoD investment will accelerate the transitions to Ada 9X and stimulate the recipients to make the additional investments to complete the products.

The DoD should plan and implement early applications of Ada 9X within DoD. Goals are to validate the quality of the Ada 9X implementations, to create favorable references for Ada 9X technology, and to provide constructive user feedback to help the vendors improve their Ada 9X products. DoD programs will have to be funded to pay the cost of their leadership in the Ada 9X transition. The Ada Technology Insertion Program (ATIP) is a model for what is required, but the required funding will be three to five times greater than previous funding of the ATIP initiative ($10M-$15M).

DEFENSE INFORMATION SYSTEMS AGENCY

Recommended Investment Priorities

- Continue Ada 9X program as planned
- DISA Ada migration plan and near-term conversion of first migration systems
- Co-funding of COTS Ada compilers, tools and bindings
- Increase funding for ATIP, and focus specifically on Ada 9X

SCRIPT FOR ADA DUAL-USE COMMITTEE BRIEFINGS

SLIDE 32 - RECOMMENDED INVESTMENT PRIORITIES

Quite a few recommendations have been presented. This slide offers a priority listing of the recommendations. The vendor panel feels that the recommendations listed are of utmost importance to the commercialization of Ada.

ADA DUAL-USE WORKSHOP SUMMARY

OCTOBER 19-20, 1993
SHERATON PREMIERE AT TYSONS CORNER
VIENNA, VIRGINIA

NOVEMBER 8, 1993

TABLE OF CONTENTS

1.0 INTRODUCTION

Approximately 175 individuals participated in the Ada Dual-Use Workshop, held on October 19-20, 1993. The workshop was sponsored by Defense Information Systems Agency's (DISA) Joint Interoperability and Engineering Organization (JIEO), Center for Information Management (CIM), Software Systems Engineering Directcrate (TXE). DISA/JIEO/CIM/TXE has been assigned the responsibility to evaluate current Ada initiatives and to align the strategies of the commercial and government sectors. This Workshop provided participants the opportunity to influence Ada commercialization strategies being developed.

The Honorable Emmett Paige Jr. opened the Workshop with a keynote address (see Appendix A for a copy of his address), which personally challenged participants to "provide [him] with the input [he] needed to make a difference." Representatives (see Appendix B for a list of attendees) from the academia, government, industry, and vendor communities addressed this challenge and recommended actions that DISA/JIEO/CIM/TXE could take to both make Ada more attractive for use within the DoD and expand its utilization within the commercial sector under the auspices of a dual-use initiative.

The purpose for commercialization is simple. DoD would like to increase the overall demand for Ada products, services, and applications, so that it can use market forces to stimulate increased investment and use of Ada by non-defense organizations. While this increased demand will create the pull toward Ada, the Congressional mandate will create the push. Both the carrots and the sticks are deemed necessary to make Ada successful.

2.0 PURPOSE

The main objective of the Ada Dual-Use Workshop was to identify ways to increase the use of the Ada programming language and related technology in both defense and commercial applications. To accomplish this goal, the workshop was organized into the following four panels: academia, government, industry and vendors. Each panel deliberated to determine ways to make Ada more attractive to both DoD and commercial users. The conclusions and recommendations resulting from the panel discussions are summarized below. A detail review of the panel session is contained in Appendices C, D, E, and F. In section five of this report the primary product of the workshop is a top-level commercialization action plan to be presented to high-level decision makers within the DoD for concurrence and funding.

3.0 REPORT ORGANIZATION

This report is organized into the following seven sections and six appendices. Section 1 introduces the reader to the workshop and discusses why it was held. Section 2 defines the goal of the workshop and the methodology used to achieve this goal. This section discusses the report's organization and points readers to sections of potential interest. Section 4 summarizes the conclusions which were gleaned as a result of the workshop. Section 5 presents the consensus recommendations and actions that need to be taken to promote Ada's use in both defense and commercial applications. Section 6 provides a summary of the results for the four panel sessions. Section 7 provides an overall summary for the report.

4.0 ADA DUAL-USE WORKSHOP CONCLUSIONS

During the workshop, the panels heard a wide variety of testimony. Many issues were discussed along with suggestions about what to do to address them. Opportunities for dual-use were also identified along with ways to take advantage of them. The relationships between these issues and the actions are illustrated in the cause and effect diagram which appears as Figure 1 (Most executives have been exposed to these fishbone diagrams in their Total Quality Management training. The technique, often associated with Kaoru Ishikawa, is referred to as a fishbone because the representation between effect and all possible causes takes the appearance of a skeleton of a fish). As Figure 1 clearly illustrates, the issues thought to be inhibiting Ada commercialization can be categorized in three main areas:

- Marketing Ada from both a technical and business point-of-view.

- Providing technical support and incentives for the expanded use of Ada within both the DoD and commercial sectors.

- Strengthening the DoD Ada mandate.

Many additional conclusions were made which are contained in the individual panel summaries. The most cited of these include:

- Ada is a true dual-use technology as demonstrated by firms like Boeing on its 777 aircraft, Motorola on its cellular phone systems and Silicon Graphics on its virtual reality simulation applications. Broader use in the commercial sector needs to be both encouraged and stimulated.

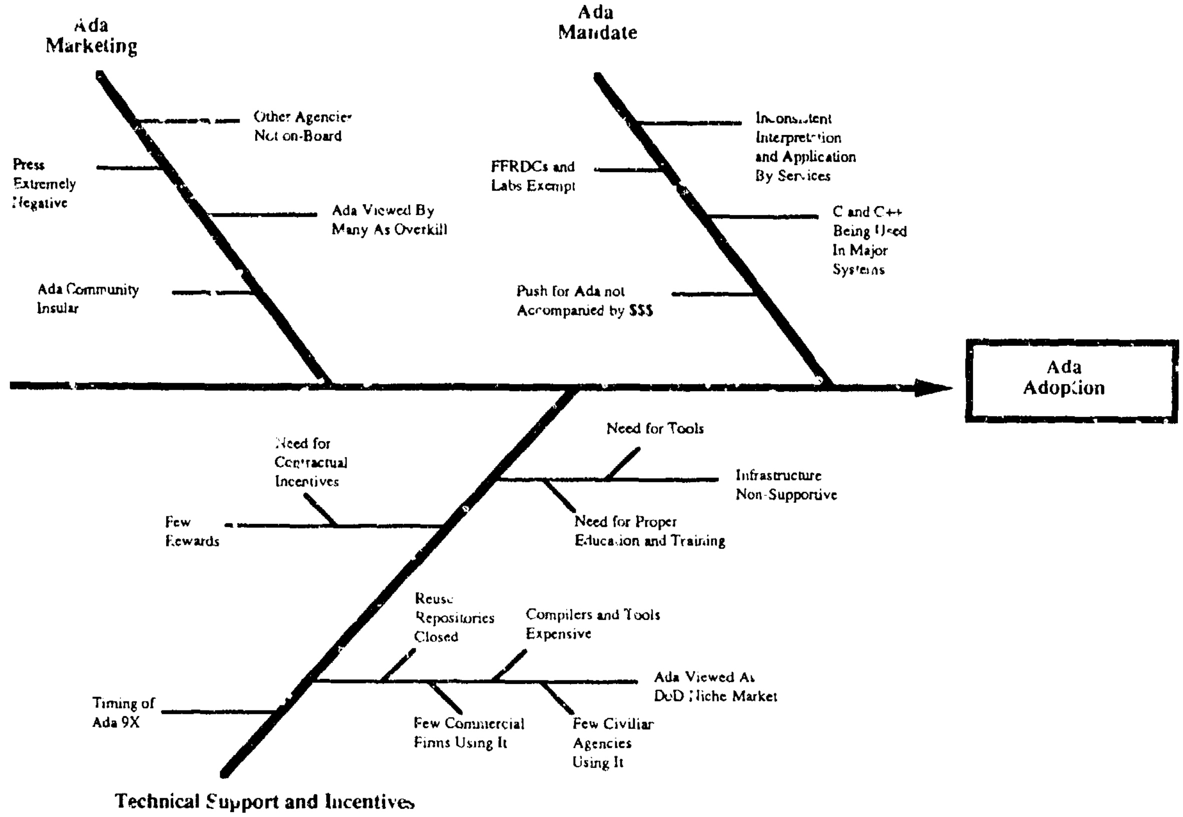

Ada Marketing
Other Agencies Not on-Board
Press Extremely Negative
Ada Viewed By Many As Overkill
Ada Community Insular
Ada Mandate
Inconsistent Interpretation and Application By Services
FFRDCs and Labs Exempt
C and C++ Being Used In Major Systems
Push for Ada not Accompanied by $$$
Ada Adoption
Need for Tools
Need for Contractual Incentives
Infrastructure Non-Supportive
Few Rewards
Need for Proper Education and Training
Reuse Repositories Closed
Compilers and Tools Expensive
Ada Viewed As DoD Niche Market
Timing of Ada 9X
Few Commercial Firms Using It
Few Civilian Agencies Using It
Technical Support and Incentives

Figure 1. Cause and Effect Diagram Showing Issues Inhibiting Ada Adoption

- There is no compelling reason for the DoD to rethink its Ada mandate. The reasons used to justify the development of Ada still exist, and the results achieved to date with its use in both the DoD and commercial sectors have been extremely favorable.

- While technical issues still exist, the primary barriers that seem to inhibit adoption of Ada continue to be educational, psychological and managerial. Ada is perceived by many as a DoD language. Its merits have neither been properly promoted nor adequately communicated to the commercial sector.

- The advent of object-oriented technology and the movement to distributed client-server architectures provides the Ada community with a rare "second window of opportunity". During the late 1980's many organizations chose C instead of Ada as they moved away from Fortran and COBOL into second and third generation computing environments. As the technology changes again, the choice seems to be between C++ and Ada 9X.

- To take advantage of this "window", additional partnerships (with industry, academia and other government agencies) are needed along with additional investments to take advantage of the positive market forces. Partnerships that exist with Department of Energy (DOE), Department of Transportation (DOT)/FAA, and NASA as well as academia need to be strengthened as the outreach program is solidified.

- A business environment conducive to continued commercial investment in Ada needs to be created. DoD should incentivize using Ada through the award, fee, and overhead structures of contracts.

- An aggressive marketing, promotion, and public relations campaign needs to be waged to improve Ada's image, communicate facts, and advance the technical and business cases needed to broaden Ada's use in the commercial sector.

- The existing support base for Ada needs to be strengthened as a first order of business. Perceived risks need to be mitigated and the Ada 9X transition needs to be accelerated to solidify the existing DoD and commercial sector support base.

- A variety of contractual (evaluation points, preferences in hirnng, etc.) and financial incentives (tax credits, changes in accounting practices, making training and tool costs allowable, etc.) should be put into place, stimulating private firms to continue investing in Ada.

- Career incentives are needed to stimulate DoD program managers and acquisition executives to emphasize Ada in their procurements, research, and in-house projects.

- Educational training and funded research initiatives are needed to motivate universities and colleges to both teach Ada and use it in their research.

- Strategic partnerships which jointly fund and develop needed capabilities (educational, industrial, governmental, etc.) should be strongly encouraged.

- DoD should encourage industry, academia, professional societies and other non-DoD groups to actively participate in its initiatives.

- DoD should give priority to investments in bindings and COTS tools which increase the attractiveness of Ada within the commercial sector.

- The Ada Joint Program Office (AJPO) has been doing many of the right things. However, as the technology transfer agent, it needs additional funding and support to make Ada commercialization a reality.

- While the policy requiring the use of Ada remains strong, it has not been consistently interpreted, applied or enforced by DoD components, FFRDCs, and research institutions. Policy deployment continues to be an issue as DoD downsizes and middle managers strive to cut costs.

- DoD needs to empower and provide DISA the means to execute this revitalization initiative.

- DISA, as DoD's executive agent for Ada, should give a high priority to moving Ada 9X from the laboratory (current state) into practice (desired state).

- DISA needs to reprioritize current Ada actions and reorient them towards increased commercial use. Their actions should be primarily focused on marketing, not technology development.

- DoD should not forget that Ada was designed and targeted for defense systems. it should resist attempts to eliminate required functionality in an attempt to gain market acceptance.

These conclusions represent a consensus of what was heard in all of the panel deliberations.

5.0 ADA DUAL-USE WORKSHOP RECOMMENDATIONS

Figure 2 represents the actions DoD executives can take to remove the barriers to Ada adoption, address perceptions, and focus on accomplishing its commercialization goal. Each action in this fishbone diagram respond to a corresponding challenge shown in Figure 1. The specific actions are briefly explained below.

To improve Ada's image, the following actions are needed:

- Build a business case and use it along with success stories to communicate Ada's excellence. Work with the Ada Software Alliance (ASA) to develop a real Ada Trade Association (Ada vendors, proponents, etc.). Provide ASA and SIGAda with the "hard data" they need to make their efforts successful. Communicate the resulting products to DoD program managers and industry executives via an aggressive public relations campaign and a Service school initiative (academies, DSMC, National Defense University, etc.). Cooperate with the National Software Repository effort mounted by the Air Force (Mosemann initiative) and provide them the data they need for their effort.

  Proposed action officer - Ada Joint Program Office (in cooperation with Service Schools, SIGAda, Ada Software Alliance and National Software Repository effort)

- Develop a marketing plan that defines the best approach for promoting Ada's use, and then aggressively follow it. Work with the newly formed Ada Trade Association, ASA and SIGAda to change the image and broaden the appeal for Ada within both the civil and commercial sectors via an aggressive advertizing and promotion campaign. Work to solidify the market base for Ada within the DoD, DoT/FAA, DoE, NASA, NATO and other supporters. Work to create the market push needed to convince software firms like Borland and MicroSoft that there is a market for Ada products and services that is broader than just the DoD.

  Proposed action officer - Ada Joint Program Office (in cooperation with the SIGAda, Ada Trade Association and Ada Software Alliance)

- Enter into cooperative agreements with academic, industry, and government agencies to stimulate use of Ada in the commercial arena using dual-use funds. Work with NSF to develop a teaching environment for Ada. Provide grants to colleges and universities which do research in and teach Ada. Utilize HBCU's as a proving grounds for Ada curriculum and courseware developments by the SEI and professional societies/groups. Enter into strategic partnerships with vendors to develop needed bindings, tools and COTS products. Work with ARPA and NIST to sponsor an Ada dual-use initiative which rewards firms that demonstrate use of Ada within commercial industry. Institute a "good idea" program to stimulate innovation and get small businesses behind the initiative. Work with DoE, DoT/FAA and NASA and get their buy into the initiative through joint sponsorship of SBIR and other programs. Utilize industry standards groups (OMG, ISO, etc.) to spread the word and get the vendors behind the initiative. Convince EIA, NSIA and other industry groups that expanded use of Ada by the commercial sector is a good idea.

  Proposed action officer - Ada Joint Program Office (in cooperation with academia, industry, industry groups, standards organizations and other government agencies)

- Develop an outreach program which communicates to the press accurate information about Ada's positive impact in both the DoD and commercial world. Provide a visible presence at major non-DoD software shows and exhibitions (Comdex, ObjectWorld, OOPSLA, etc.). Cooperate with SIGAda to reach out to other interest groups (Medical SIG, Numerics SIG, etc.). Advertize in major publications. Hold quarterly briefings to industry. Prepare press releases which publicize Ada successes. Write articles that extol the virtues of Ada within the commercial sector. Hire a leader in the field to act the spokesperson for the industry. Influence the influence makers and get them to speak positively about Ada within public forums and other media events.

  Proposed action officer - Ada Joint Program Office (in cooperation with SIGAda)

- Encourage the exchange of information by broadening the role of the Ada Information Clearinghouse (AdaIC). Currently, they provide information to firms and organizations within the defense sector. Work with them to go outside the box. Establish on-line bulletin boards that more broadly enable exchange of information and ideas. Increase the frequency of newsletter publication and publish monograms and white papers. Encourage industry groups, professional societies and trade associations to distribute Ada information in their mailings and publications.

  Proposed action officer - Ada Joint Program Office

To stimulate the incentives which will make Ada commercialization a reality the following actions are needed:

- Reward all stakeholders in the Ada initiative by making it in their best interests to use Ada. Work with the Comptroller and legal staffs to provide financial reasons why DoD suppliers (universities, vendors, contractors, etc.) should use Ada. Incorporate Ada competence into the SEI CMM. Permit Ada training and tools as allowable costs in contracts. Work with personnel to make sure that Ada competence is rewarded via improved funding, educational opportunities, and chances for promotion. Implement a Malcolm Baldridge-like award for Ada excellence. Emphasize Ada issues in IR&D, pre-award and post-award reviews.

  Proposed action officer - Ada Joint Program Office (with support from Software Executives, Comptroller, Legal and Software Engineering Institute when needed to address commercialization efforts)

- Improve the infrastructure available for DoD program managers by providing the tools, bindings, COTS products, guidance, educational and training opportunities, and support needed to make it easy to adopt Ada on DoD and (potentially) commercial projects. Emphasize Ada education in the Service schools (DSMC, etc.), HBCU's, the SEI and university programs sponsored by the DoD . Make sure that other DoD initiatives (MIL-STD-SDD, ICASE, etc.) support the Ada initiative. Open the DoD reuse libraries and use them as a distribution mechanism for quality Ada reusable components and information.

  Proposed action officer - Ada Joint Program Office (in cooperation with DoD Program Managers for ICASE, MIL-STD-SDD and Software Reuse Initiative)

- Accelerate needed Ada 9X support by entering into strategic partnerships with tooi vendors and training providers. Demonstrate the ability of Ada 9X in relationship to DoD user needs and the "window of opportunity" presented by object-oriented technology via demonstrations, ATIPs, SBIRs and funding support for pilot projects. Insist on measurement of success in everything you do. Enter into partnerships outside of the DoD arena that permit the advantages of Ada (both technical and business) to be demonstrated convincingly to doubters and critics. Focus on technology transfer because that's where the action really is. Move Ada 9X into practice according to a well thought out and coordinated transition plan.

  Proposed action officer - Ada Joint Program Office (in cooperation with Ada9X Program Manager)

- Innovate to increase the commercial use of Ada. Open the reuse repositories to provide Ada users with access to quality tools, bindings, and COTS products. Reduce the costs of compilers and tools to startup firms and universities using subsidies, grants, and free software. Permit subsets and supersets of the language for teaching and research purposes. Cooperate with ARPA and NIST to use dual-use funds to stimulate commercialization. Cooperate with industry and academia, and jointly fund technology transfer mechanisms. Emphasize Ada in our research programs by encouraging Principal Investigators to look at software issues as part of their grants.

  Proposed action officer - Ada Joint Program Officer (with support of Secretary of Defense, ASD(C3I) and DoD Software Executives when needed to address InterAgency issues)

To enforce the mandate, the following actions are needed:

- Continue to commit to the Ada mandate in public forums and events. Issue the revised version of the language policy (DoDD3405.1) as soon as possible and add teeth to it through the acquisition review cycle. Eliminate any inconsistency and variation within DoD components by providing interpretation guidance.

  Proposed action officer - Ada Joint Program OfEce (with support from ASD(C3I) and DoD Software Executives when needed to address procurement law issues)

- Enforce the Ada mandate by supporting only Ada projects. Provide detailed guidance on waiver policies and enforce it by having staff review all pending solicitations and contract renewal packages for compliance. Have Software Executives work within their organizations to deploy policy within 30 days. Have Inspector General conduct audits to spot check compliance. Withdraw funding support for those programs which fail to comply with policy and direction.

  Proposed action officer - Ada Joint Program Office (in cooperation with DoD acquisition staffs, Comptroller, Inspector General and Software Executives)

- Motivate ARPA, FFRDCs (e.g. SEI and others), Service labs, and Service schools to use Ada in their research and technology demonstration projects. Have the SEI incorporate Ada into the CMM. Require all FFRDC's (Aerospace, IDA, MITRE, SEI, etc.) to advocate and use Ada internally as well as externally in their research and advanced development programs. Permit Ada subsets and supersets to encourage development of inexpensive tools for teaching of Ada within Service schools. Permit use of invalidated Ada 9X compilers prior to issuance of the standard to encourage learning and use of the language.

  Proposed action officer - Ada Joint Program Office (with support from ASD(C3I) and DoD Software Executives when needed to address InterAgency and FFRDC contractual issues)

- Require contractual requirements for the C and C++ programming language to be justified prior to any allocation of funding. Review all solicitations and contract renewal packages to determine where C and C++ have been required instead of Ada . Have Software Executives work within their organizations to review justifications for use of the C and C++ programming language within 30 days. Forward justifications to the ASD (C3I) for staff review within 60 days.

  Proposed action officer - Ada Joint Program Office (with support from ASD(C3I), DoD Software Executives and their acquisition staffs when needed to address compliance issues)

- Working with ARPA and NIST, develop a dual-use investment strategy. Task the Ada Joint Program Office (AJPO) to prepare a plan of attack within the next 45 days. Execute the approved plan, monitor results, and manage the initiative proactively (currently viewed as reactive). Empower DISA to manage the initiative by transferring funding, staff and all applicable program elements from DDR&E as soon as possible. Manage the Ada initiative as a program of major importance.

  Proposed action officer - Ada Joint Program Office (with support from ASD(C3I) and DoD Software Executives when needed to address InterAgency issues)

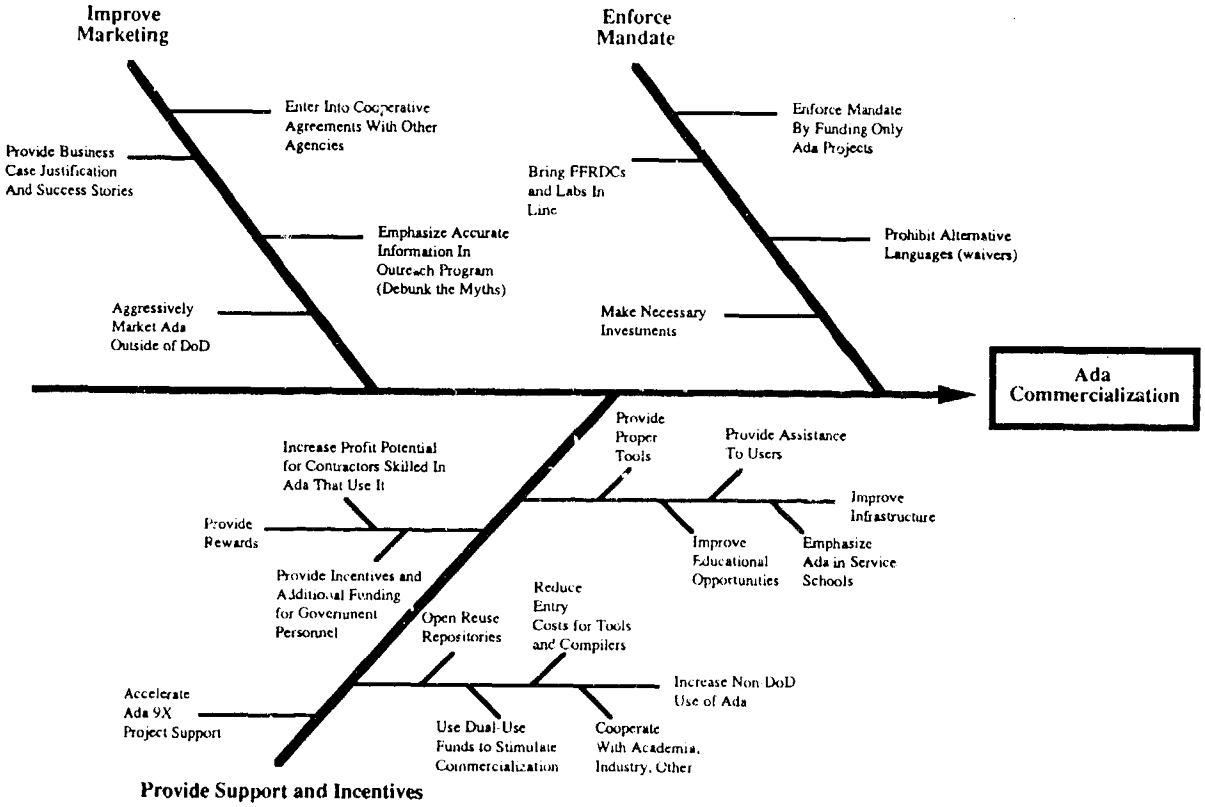

Improve Marketing
Enter Into Cooperative Agreements With Other Agencies
Provide Business Case Justification And Success Stories
Emphasize Accurate Information In Outreach Program (Debunk the Myths)
Aggressively Market Ada Outside of DoD
Enforce Mandate
Enforce Mandate By Funding Only Ada Projects
Bring FFRDCs and Labs In Line
Prohibit Alternative Languages (waivers)
Make Necessary Investments
Ada Commercialization
Provide Proper Tools
Provide Assistance To Users
Improve Infrastructure
Improve Educational Opportunities
Emphasize Ada in Service Schools
Increase Profit Potential for Contractors Skilled In Ada That Use It
Provide Rewards
Provide Incentives and Additional Funding for Government Personnel
Open Reuse Repositories
Reduce Entry Costs for Tools and Compilers
Increase Non DoD Use of Ada
Accelerate Ada 9X Project Support
Use Dual-Use Funds to Stimulate Commercialization
Cooperate With Academia, Industry, Other
Provide Support and Incentives

Figure 2. Ada Commercialization Strategy

To implement these initiatives, and achieve successful Ada commercialization, a six step process is recommended. The process is illustrated in Figure 3 and explained as follows:

1. **Plan** - Develop a preliminary plan of attack by November 24, 1993.

2. **Solidify** - Solidify the existing base by reconfirming the mandate, aggressively pursuing Ada 9X, and implementing an effective Ada marketing/outreach program within the DoD.

3. **Extend** - Extend the existing base by entering into cooperative agreements with academia, industry, and other government agencies to develop needed Ada 9X tools, bindings and COTS products, and to provide guidance and support under the auspices of the dual-use program. Extend the role of the AdaIC, if warranted.

4. **Apply** - Apply products as they evolve using pilots to demonstrate technologies. Use successes to breed successes as part of an aggressive marketing/promotion campaign.

5. **Amplify** - Amplify efforts by marshalling commercial forces to synergistically generate results. Build momentum, channel effort, and act as a catalyst.

6. **Reiterate** - Proactively manage the initiative, reiterating when necessary, and pushing forward when warranted. Get things moving and keep them under control.

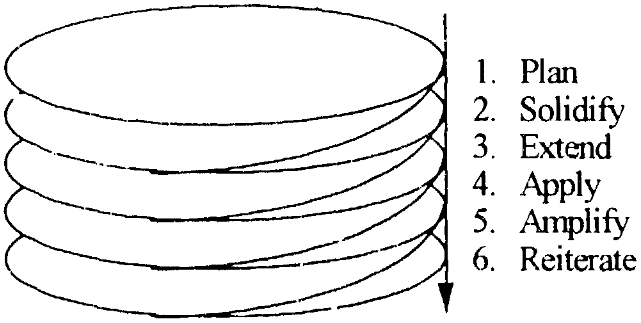


Figure 3. Commercialization Process

Based upon the workshop, we have identified ten near-term targets of opportunity as shown in Table 1. It is recommend that the dual-use committee provide us with direction to proceed with taking advantage of these within the near-term.

An initial assessment of potential investment is presented in Table 2. Although more details are needed, we believe funding at approximately $20 million per year for five years is required to make Ada commercialization a reality. The DoD dual-use program within ARPA is the primary source for these funds.

Of course, any initiative of the magnitude presented needs to be managed as a major program within the DoD. Therefore, we recommend that the Ada program be elevated and given its program element within the DoD budget. Reporting and management cognizance for the program should be given to DISA.

TARGET	OPPORTUNITY
1. ARPA dual-use programs	Emphasize Ada commercialization in dual-use programs
2. NIST and NSF	Make Ada commercialization central to their core technology and dual-use programs via cooperative agreements and joint funding
3. Service Schools	Teach Ada in undergraduate curriculum
4. DSMC and NDU	Educate PMs/PEOs in Ada business case
5. HBCU	Use as a proving ground for curriculum and courseware
6. SEI	Incorporate Ada in CMM, curriculum and products
7. FFRDCs and Labs	Perform research in Ada at a level at least equivalent to C or C++ and work tool/environment issues for Ada 9X
8. Ada 9X Program Office	Accelerate near-term support for programs using Ada 9X
9. DISA/JIEO	Demonstrate the technology on migration systems
10. SBIR	Reemphasize Ada in next year's solicitation

TABLE 1 - TARGETS OF OPPORTUNITY

INVESTMENT	OPPORTUNITY	AMOUNT
1. Improve Marketing	• Business case development • Marketing Plan • Outreach program/promotion campaign • Image improvement campaign	$ 1,000,000
2. Provide incentives and support	• Expanded AdaIC role • Strategic tool partnerships • "Good idea" program • Open reuse repositories • SEI CMM modification project • Pilot projects using Ada 9X • Vendor subsidies for Ada 9X products • Subsidies for DoD research in Ada • Expanded ATIP program • Production Ada teaching environment • University grant program	$15,900,000
3. Enforce Mandate	• Policy interpretation and deployment	$ 100,000
4. Provide leadership and management	• Public relationship program • Program management	$3,000,000

TABLE 2 - INVESTMENTS (FY94)

6.0 PANEL SUMMARIES

The panel sessions provided participants with the opportunity to present their particular viewpoints and recommendations for increasing Ada utilization and commercialization. Participants focused their efforts on answering questions designated for each panel within the four disciplines: academia, government, industry, and vendors. Participant briefings provided the basis for the panelists' findings, conclusions and recommendations. The subsections that follow provide a brief summary of the results for each panel. Appendices C, D, E and F of this report provide detailed panel findings, conclusions, and recommendations for each of the questions the participants addressed.

6.1 ACADEMIA PANEL SUMMARY

The Academia Panel Session provided participants with the opportunity to present their particular viewpoints and recommendations for initiatives to increase the use and teaching of Ada. Briefings were presented by members of the academic community, military academies, other educational institutions, and training vendors. Participants focused their efforts on answering four questions which provided the basis for the panelists' findings, conclusions and recommendations (Refer to Appendix C, Ada Dual-Use Workshop - Academia Panel Proceedings for the supporting data which includes a summarization of the panel findings, conclusions, and recommendations for each question). The questions consisted of the following:

1. **What actions can we take to make Ada one of the primary teaching languages within major colleges and universities?**

2. **What initiatives can we mount to increase Ada's appeal, adoption, and popularity within the academic community?**

3. **What investments can we make to increase the commercial use of Ada?**

4. **What steps can we take to increase the pool of entry level personnel with Ada skills, knowledge, and experience?**

The key topics derived from the two day academia panel session are summarized as follows:

a. Leveraging academic accomplishments with Ada 83 as a foundation for subsequent improvements
b. The relationship between Ada and related software engineering activities
c. Availability of low cost environments, bindings and tools
d. The role of educational training and research in stimulating Ada dual use
e. Consistency of the DoD commitment to Ada

The academic panel concluded that this two day workshop produced a useful discussion of key issues and possible approaches for institutionalizing Ada 9X within academia and thereby contributing to the commercialization of 9X. However, continuing dialogue and cooperation will be required if academia is to be involved as a full-fledged partner with government, industry and vendors in achieving the goals identified for this workshop.

Government, industry, vendors and academia have made notable progress toward goals complementary to those of this workshop. The infrastructure and accomplishments of this past work should be leveraged as a foundation for subsequent improvements.

There is already a good baseline of Ada-literate teachers, computer science texts, free or inexpensive compilers, Ada-related academic organizations, and non-defense successes.

The Academic panel has concluded that we need MORE of what has made Ada succeed, NOT whole new efforts!

The academic panel recommends the following actions be immediately undertaken:

a. Restore/initiate consistency in DoD's treatment of Ada and Software Engineering.

b. Expand an existing forum, ASEET (Ada Software Engineering Education and Training), to serve the needs for timely information and technology exchange among all educators that are stakeholders in Ada 9X.

c. Low cost/free environments, tools and other resources supporting instruction and research in and with Ada 9X should be made available to educators and students as soon as possible.

d. Increase Ada instruction and research support. NSF-style faculty development workshops should be made available to facilitate the expansion and integration of Software Engineering and Ada 9X instruction and research in academia. In addition, DoD should sponsor an initiative to effect curriculum improvement that covers and integrates issues across the software engineering life cycle.

6.2 GOVERNMENT PANEL SUMMARY

The Government Panel Session consisted of briefings presented by members of the software engineering community as a whole, not just the Ada community. Participants consisted of users, project managers, and senior executives within the DoD, as well as other non-DoD agencies throughout the federal government sector. Software support contractors were also represented. Refer to Appendix D, Ada Dual-Use Workshop - Government Panel Proceedings for the supporting data which includes a summarization of the panel findings, conclusions, and recommendations for each question. Three questions were carefully considered by all the participants:

1. **What action can we take to make Ada the primary tool for developing software systems within your government organization?**

2. **What investments can we make to increase the commercial use of Ada?**

3. **What can we do to increase Ada's appeal, adoption, and popularity within your software community?**

The key topics derived from the two day government panel session are summarized as follows:

a. Marketing
b. Training and Education
c. The DoD Mandate
d. Bindings
e. Tools
f. Incentives
g. Reuse Libraries
h. Validation
i. Research

Based on the briefings that were presented by the participants, the government panel summarized their recommendations as follows:

a. Marketing - The DoD should provide information that substantiates the benefits of using Ada during a given project's Life Cycle Management (LCM) process. This includes providing examples that are considered major Ada success stories. In essence, the DoD should advertise "success stories" within publications whose readership includes the audience that we are trying to reach - particularly, the business world executive (examples of such publications include Business Week, Forbes, Datamation, etc.).

b. Training and Education - The DoD should develop videos in conjunction with other federal agencies that identify existing programs, provide statistics, teach the lessons that have been learned, and describe a listing of available sources of information for instructional help on Ada. The Ada community as a whole needs to educate the universities and colleges about Ada, and support and proliferate its use. The DoD needs to sponsor efforts at the elementary, secondary and community college education levels to ensure continued and successful utilization of the language.

c. DoD Mandate - The DoD needs to clarify and promulgate the Ada mandate for the defense industry. DoD also needs to better inforce the Ada policy. This will support the Dual-Use initiative and will refute the claim that "the mandate is not directed at me". The DoD needs to lead by example.

d. Bindings - The Ada community needs to foster an effort that focuses on Ada bindings through identifying required bindings, working with industry and NIST to develop and publish standard binding specifications, narrowing the choices among competitors, stimulating comparison efforts, publishing specifications, supporting prototype implementation through the subsidy of binding efforts, interfacing with numerous languages at every level, and finally, standardizing the appropriate specifications.

e. Tools - Ada tools need to be supported. Ada needs to have a rich set of tools that support new language versions such as Ada 9X, just like "C" does. AdaSAGE needs to keep current with Ada 9X. In addition, DoD needs to publicize the availablity of Ada tools through the I-CASE program.

f. Incentives - The DoD and other federal agencies need to recognize Program Managers of Ada intensive projects with a special conference/gathering that rewards their involvement and efforts. The reward could be provided through the OASD(C3I). Spot awards could possibly be made throughout the DoD and federal government.

g. **Reuse Libraries** - The community as a whole needs to provide reusable Ada libraries for all environments, domains, and platforms.

h. **Validation** - The DoD needs to allow superset/subsets of Ada. This would help Ada reach colleges and universities. Also, the DoD should look at allowing non-validated Ada 9X compilers for non-critical systems until validated ones reach the market. This would speed the process of Ada 9X reaching the Ada community.

i. **Research** - The DoD needs to subsidize more research in Ada and especially Ada 9X. The funding for research could be provided through ARPA dual-use funds

6.3 INDUSTRY PANEL SUMMARY

The Industry Panel Session consisted of briefings presented by members of the software engineering community as a whole, not just the Ada community. Participants consisted of users, project managers, and senior executives throughout the industry sector, including both government contractors and commercial markets. Refer to Appendix E, Ada Dual-Use Workshop - Industry Panel Proceedings for the supporting data which includes a summarization of the panel findings, conclusions, and recommendations for each question. Participants briefed panel members on the following questions:

1. **What actions can we take to get Ada accepted as the preferred language for development of software within your organization?**

2. **What investments can we make to increase the commercial use of Ada?**

3. **What can we do to increase Ada's appeal, adoption, and popularity within your software community?**

The key topics derived from the two day industry panel session are summarized as follows:

a. Enforcement of DoD Mandate
b. Business case for Ada
c. Bindings and Tools
d. Marketing
e. Education and Training
f. Reuse Libraries
g. Incentives

Based on the briefings that were presented by the participants, the ˈndustry panel summarized their recommendations as follows:

a. **DoD Mandate** - The DoD should strengthen the Ada mandate by enforcing the waiver process and adhering to public law. Reinforce the DoD's language policy to include high integrity, research and development and information systems where life cycle cost issues are paramount. Provide incentives for those firms that have proven Ada capabilities on legacy systems, or provide awards for DoD Program Managers and firms who demonstrate Ada competence during contract performance.

b. **Business Case** - In order to place a business focus on Ada plans and strategies, the DoD should cooperate with existing efforts (SIGAda, Ada Software Alliance, etc.) to substantiate Ada's excellence. A solid business case and briefing should be developed for Ada with global statistics that represent quantitative data on comparative performance (cost, schedule, quality, etc.) of quality issues throughout the life cycle. In addition, the DoD should capture success stories, leverage lessons learned, and focus discussions on why Ada was selected and what benefits resulted.

c. **Bindings and Tools** - There is a need for more bindings and COTS tools. The DoD should establish partnerships with Ada vendors to jointly develop high demand items on a cost-sharing basis (Government could take limited rights; contractor could get marketing rights). Additionally, the DoD should use reuse libraries as a distribution mechanism for high demand items on a fee-for-service basis. Finally, the DoD's strategy should look toward coexistence, not toward exclusive use of Ada. A coexistence approach with other languages (C and C++ based products), in their technically appropriate area of application, should be evaluated.

d. **Marketing** - The DoD should define a marketing strategy for the commercialization of Ada. This should include defining the market and market needs for Ada within the DoD. The DoD's marketing strategy should include a plan for improving the outreach of Ada. This could be accomplished by establishing a marketing objective for the AJPO that includes interacting with professional societies, FFRDCs, trade associations, other government agencies, universities, and industry. The DoD should also reconstitute the Ada Board to act as a sounding board for new ideas. The focus of the board should be to develop a plan to move from a technology-driven strategy to one that is market-driven. In addition, to solidify the existing Ada base, the DoD should view migration systems within DISA as a potential opportunity to spotlight Ada.

e. Education and Training - The DoD should stimulate educational use of Ada by permitting language subsets and supersets and by providing grants and other forms of assistance to high schools, universities, and colleges for Ada-based software engineering training. Preference in hiring should be given to Ada trained software entry level engineers and programmers within the DoD. A teaching environment and low-cost compilers for Ada-based software engineering training should also be supported by the DoD. In addition, the DoD should place a greater emphasis on training government personnel. This could be accomplished by providing videos and Multi-Media based educational software for self-paced management and programmer education.

f. Reuse Libraries - The community as a whole needs to provide reusable Ada libraries for all environments, domains, and platforms.

g. Incentives - The DoD should provide incentives that will motivate investors to renew their Ada initiatives, and should stimulate industry to develop and recognize core competency in Ada. The DoD should establish joint investments and strategic partnerships with other government agencies, universities and colleges, professional societies and trade associations, and dual-use programs (NIST, ARPA, NSF, etc.). Incentivize contracts for Ada use.

6.4 VENDOR PANEL SUMMARY

The Vendor Panel Session gave participants the opportunity to present their particular viewpoints and recommendations for increasing Ada utilization and commercialization. Briefings were presented by members of the software engineering community as a whole, not just the Ada community. Participants consisted of users, project managers, and senior executives throughout the vendor sector including both government contractors and commercial markets. Refer to Appendix F, Ada Dual-Use Workshop - Vendor Panel Proceedings for the supporting data which includes a summarization of the panel findings, conclusions, and recommendations for each question. The following three questions were posed to all participants:

1. **What can we do to increase the market for Ada goods and services within the non-defense sectors?**

2. **What investments should we make and initiatives should we take to increase the commercial use of Ada?**

3. **What can we do to increase Ada's appeal, adoption, and popularity?**

The key topics derived from the two day vendor panel session are summarized as follows:

a. DoD commitment to use Ada
b. Awareness of Ada in the commercial sector
c. Providing a business environment that is conducive to vendor investment
d. Role of Ada 9X in promoting dual use

The vendor panel determined that the DoD and Ada vendors can be effective partners in increasing the market for Ada goods and services in dual-use applications. This can be accomplished by focusing on specific market segments and capitalizing on their respective strengths.

The panel concluded that a particular target of opportunity is the reengineering of Information Systems into architecturally designed Ada components which can be ported easily to run in diverse environments ranging from mainframes to open client/server configurations and personal computers. Ada's technical benefits, especially portability and facilitation of reliable component level reuse, are especially valuable in these applications.

The four highest priorities for DoD action are:

- To demonstrate a consistent commitment to use Ada for DoD applications.
- To partner with industry in promoting an awareness of Ada's benefits.
- To create a business environment which is conducive to private investment.
- To accelerate the availability of high quality Ada 9X products.

The DoD should be aggressive in implementing these high priority actions. A detailed analysis of each action area, including the findings which motivate the recommendation, conclusions, and specific implementation steps is provided in Appendix F.

Vendors must also do their part. They must provide high quality products and services to meet DoD requirements, and they must be aggressive in promoting their products and services within non-Defense sectors.

7.0 ADA DUAL-USE WORKSHOP SUMMARY

The Ada Dual-Use Workshop provided an opportunity for academia, government, industry, and vendors to influence the future of Ada commercialization. The results of the Workshop are summarized in this report, along with the detailed findings, conclusions and recommendations of the four panels. In addition, an initial high-level action plan has been developed. These materials will be presented to the Ada Dual-Use Workshop Committee headed by the Honorable Emmett Paige, Jr., Assistant Secretary of Defense for Command, Control, Communications and Intelligence (ASD/C3I) and Chaired by LTG Alonzo Short, Director, Defense Information Systems Agency, on November 8, 1993. A tentative agenda for that meeting is as follows:

Introduction and Welcome	Mr. Russomano	10 minutes
Academia	Dr. McKay	20 minutes
Government	Dr. Fischer	20 minutes
Industry	Dr. Straeter	20 minutes
Vendors	Mr. Carlson	20 minutes
Summary	Mr. Russomano	30 minutes
Break		15 minutes
Dual-Use Committee Discussion		45 minutes
Dual-Use Committee Recommendation		1 hour

The Ada community views the Workshop as a success. The community's testimony is reflected in the findings, conclusions and recommendations contained within this report. In addition to being pleased with Mr. Paige's strong showing of support, the community hopes that positive action aimed at commercialization will result as a direct consequence of these hearings.

APPENDIX A

ADA DUAL-USE WORKSHOP
OCTOBER 19-20, 1993

SPEECH BY
THE HONORABLE EMMETT PAIGE, JR.

ASSISTANT SECRETARY OF DEFENSE
COMMAND, CONTROL, COMMUNICATIONS AND INTELLIGENCE

November 8, 1993

Defense Information Systems Agency
Joint Interoperability and Engineering Organization
Center for Information Management
Software Systems Engineering Directorate

The Honorable Emmett Paige Jr.'s speech was taken literally from transcripts provided by court reporters.

General Short, I want to thank you for your remarks. I think back when we started with the Information Systems Command and reach back to find a tall army colonel that was supposed to be my deputy at Adelphi and promoted him to brigadier general, then moved him to my command, my new command. Then I sent him up to Monmouth and then to ISAC. And he was supposed to do all that was needed to be done with Ada, he and General Salisbury. And I don't imagine Salisbury is coming today. But what I should do is put you and Salisbury up here and get me a belt, a leather one, and whip on you all a little bit, because if you all had worked harder at it, and if you all had really believed in it then, we wouldn't be here today.

So I don't want you leaving here when you retire from the army and go out and start talking and saying bad things about Ada. So if you have anything bad to say, you better put them on the table now, because when you leave the service, we can always call you back if you don't act right.

Now, first off, before I even get to my prepared remarks, I want to tell you that I'm not a supporter of Ada for the sake of Ada. If I didn't believe in it, if I didn't think it was necessary, I wouldn't be here, and never would have been a supporter of it from 1979, when I moved across the highway at Fort Monmouth to find out what it was all about.

And I can assure you that the whole reason for Ada, in the first place, still exists. The need today is probably greater than it ever was, greater than it was then. And a lot of people out there that didn't even believe in automated systems back in those days are now believers in automated systems. So I say the need for Ada is even greater than it was in the beginning. And that's my vision and my view of the world.

Since its introduction, DoD and the software engineering community, I believe, have benefitted greatly from Ada. With the features that Ada 9X promises to bring to the table, things will get even better. While we took some early heat because of lack of quality and validated compilers in the early days, the results to date show quantitatively that Ada makes sense, both technically and from a business point of view.

Needless to say, the policy official in DoD that's responsible for C-4IA is firmly committed to the Ada strategy. Any speculation that DoD is wavering on the commitment to Ada is wrong. Based upon the results that we've seen today, we have no compelling reason to rethink our Ada strategy. This strategy uses Ada as a kingpin to bring a software engineering discipline into DoD. And I'm sure that most of you, if not all of you people that are here today, recognize the need to make software engineering really a discipline. In my humble view, we have held Ada too tightly in the past. And as someone said, it's just not getting enough air. And we've got to loosen our grip.

Based on my experience within the Department of Defense, combined with my recent experience in the private sector, there appears to be a common misconception that Ada has not received the level of support that is necessary to ensure its acceptance within all of the Department of Defense and the commercial sectors. I know for a fact that there are many private firms that have embraced Ada. For example, I've read that Silicon Graphics is using Ada in some of its virtual reality simulations. And I'm sure that they would not be using Ada if they did not think it was the best language to use for the particular purpose. NASA and FAA are using Ada in their major projects. And, certainly, we will not blame all of the computer problems and other disasters that NASA has suffered in the past few years to their use of the Ada language. Likewise, we cannot blame FAA's use of Ada for the program delays that they've encountered. Many firms overseas, like Ferrani, Nippon, are using Ada with positive results.

However, while I have not witnessed these examples firsthand, I nonetheless feel that Ada has not penetrated the commercial sector to the degree that we, within DoD, had hoped. The Ada market is perceived as a niche market by many of the vendors that I talk with. And, personally, I believe there is a lot of truth to this assessment. That's one of the reasons why we are here this morning. And that's one of the issues that we need your help in addressing. We've got to increase the appeal of Ada outside of the Department of Defense, and outside of the federal government. We will, forever, have to carry the burden of the industrial Tower-of-Language-Babble on our shoulders if we don't do something to improve the utilization of Ada throughout our sector. While we are not afraid to continue our investments, the best of all possible worlds would be one in which the market itself caused others who would spend their money to enhance their market share of a larger Ada market internationally. The pull needs to accompany the push or else our chances of success in the future will be limited. I would like to see U.S. companies leading the world in building jobs in this country, producing applications using the Ada language.

In my view, we in the federal government have tried to prime the pump with Ada, but so far we've failed to find the answer to cause the commercial sector in the United States to pick up the Ada baton and use it for their own purposes. A few years ago, I used to say that the day that we see IBM adopt Ada for their commercial systems will be the day that we can say that Ada has arrived. Today, I will say that the day that we see Microsoft and Borland using Ada, we can say that Ada has arrived.

There are other issues, as well. A lot of people still believe Ada is an overkill, that it's slow, that it's non-responsive. But I believe that we've come a long way in the area of performance. All of the benchmarks that I've heard about indicate that most Ada applications run as fast and are more robust than their counterparts in other languages. With regard to richness, this has never been a handicap to those properly trained in software engineering in Ada. As a matter of fact, those people that are trained in Ada love the capabilities that the language provides to them.

The availability of quality compilers, tools and environments, including the linkages, has also been raised as an issue in the past. And, again, I believe we've come a long way in this area in the past ten years. Those of you who are old enough to remember the early compilers will recall, as I did, the inefficient beast that consumed whole machines. Today, compilers can run effectively on work stations on your desktop. And when combined with ICASE, hopefully, and I repeat, hopefully, we will finally get the quality environment that we in the community have all been looking for. I'm told that Rational will provide their new environment, Tool Set, free of charge to educational institutions across the country. Certainly, that will enhance the training of people in the Ada language in many curriculums. And that, too, will hopefully help institutionalize the language throughout the industry.

To paraphrase Mark Twain, "The rumors of Ada's demise have been greatly exaggerated." As I said earlier, Ada is alive in the Department of Defense and within segments of our commercial industry. But being there is not enough. We want it to thrive. We've got to take Ada to a higher level. And we're trying to build strategies within DoD in the private sector to do just this. However, as the questions that we sent to you indicate, we are not presumptuous enough to assume that we have all the answers. We need your input. And we need your recommendations. That's the primary reason why we're here today.

Our strategy focuses on strategic partnering, cooperative efforts, integration and bindings. And we are considering a more aggressive commercially-based approach. We are looking at many carrots and maybe a few sticks, as Jerry said. And we need to make Ada easier to use than not to use. We need to provide those, in a right-sizing environment, compelling reasons for using Ada. What we have in mind is marrying Ada with several of our other initiatives. Just as an example, we are considering providing bindings to C and C++, so that Ada programmers can access proven, reusable software components in other languages through Ada interfaces; providing educators with a quality teaching environment for Ada, as I mentioned a few moments ago, also will help. We also need to better market Ada and get rid of those misconceptions that inhibit commercial use of this powerful technology.

The users of Ada in the government contracting community and in the commercial sector is, and will continue to be, our greatest, most effective marketing source. You must talk about your successes as well as your failures. Talk about your problems, any problems, any successes that you have in using the Ada language. We simply have not done enough talking about the utilization. We intend to help you do that better than we have in the past. And to implement change, we need to ensure that our investment strategy spends money on the things that will really make a difference.

For me, again, that's a primary output of this dual-use workshop. What I really want to do is to take your recommendations and run with them. I want to make your job and our job easier.

Let me tell you what's going to happen as a result of your input. On November 8, my executive-level committee will meet to review all of the findings and recommendations that come out of this workshop. We will prioritize your suggestions and task the director of the Ada joint program office to incorporate those in the plan for this year.

Your challenge is to provide me with the input I need to make a difference. I'm counting on you. And you can count on me to keep the interest and the enforcement of the OSD theater policy on the front burner.

I certainly intend to enforce it. And those people who tell you that they can get waivers, remind them about Stan Fins, if you will. The then-assistant secretary of the army for financial management and Clyde Jeffcoat, both two tough guys, were determined that they were not going to use Ada. And although the assistant secretary of the army was, I guess, supposed to be my boss, he didn't take me on. So I'm going to take on anybody that comes in and asks for a waiver. And if General Short gets one, and he doesn't send it up to me, well, then, he'll retire early.

As I frequently say, I did not come back to OSD to play any games. I absolutely did not. I came here to try as hard as I can to be a strong stalwart in getting on with the job that must be done in the C4I business across corporate DoD. Business, as usual, will not get the job done. Working together, we can make things happen.

That's enough of a pep talk. I would now like to share my broader vision with you. I see a time, during my tenure, when DoD software engineers use Ada not just by mandate but by preference.

They have the tools at their fingertips and the processes in place to make a difference when they have open access to quality, reusable software components from our reused repositories and reap the benefits from their controlled use.

I see commercial software houses investing in owning Ada because the market is there and to do otherwise would be a competitive disadvantage.

I see a steady stream of Ada-trained software engineers flowing out of our colleges and our universities.

I see the DoD being viewed as leaders, not followers, in software engineering and technology exploitation.

I see software costs continuing to go down and quality continuing to go up.

And, most important, I see and desire compelling managers to select Ada because of business as well as technical justification.

That's my vision. And I thank you for your indulgence and getting up early enough this morning to come in here to see what I was going to say to you. I know that you've donated your time, and in many cases, your travel funds, to assist us this week in this effort. For that, I want to say thank you. I'm especially grateful to you. I'm pleased with the turnout.

APPENDIX B

ADA DUAL-USE WORKSHOP
OCTOBER 19-20, 1993

ATTENDEE LIST

November 8, 1993

Defense Information Systems Agency
Joint Interoperability and Engineering Organization
Center for Information Management
Software Systems Engineering Directorate

TABLE OF CONTENTS

1.0 REGISTRATION STATIISTICS

Attendees Registered Onsite:		174
Panel Chairs	4	
Panel Facilitators	9	
Panel Members	19	
Panels:		
Academia	22	
Government	25	
Industry	28	
Vendor	23	
Observers	31	
Press	2	
Committee	4	
Staff	7	

2.0 LIST OF ATTENDEES

Adams, Elizabeth Dr.
Hood College

Aha, Stephanie Ms.
EVB Software Engineering, Inc.

Aharonian, Gregory Mr.
Source Translation Optimization

Arnette, Joanne Ms.
Defense Information Systems Agency

Bacarro, J. Paul Mr.
Defense Information Systems Agency

Baker, Roger Mr.
Verdix Corporation

Baldwin, Fred Mr.
Intermetrics, Incorporated

Balfour, Brad Mr.
SofTech, Incorporated

Banta, Richard Mr.
Westinghouse Electric Corporation

Barrett, Johnny Mr.
Texas Instruments, Incorporated

Battey, Craig Mr.
Genesis Software, Incorporated

Battista, Mary Ms.
Ada Dual-Use Workshop Staff,
Electronic Data Systems, Incorporated

Bedar, George Mr.
U.S. Naval Academy

Berg, Richard Mr.
Defense Mapping Agency

Berman, Debbie Ms.
Ada Dual-Use Workshop Staff,
Electronic Data Systems, Incorporated

Bohn, Bruce Brig Gen, US Air Force
Defense Information Systems Agency

Bolz, Richard Mr.
Ventura Technology

Bond, Jan Mr.
Ventura Technology

Bowers, Dennis Lt Col, US Air Force
Defense Information Systems Agency

Boyd, Stowe Mr.
Verdix Corporation

Brown, Linda Ms.
OASD (C3I)

Burneskis, John Mr.
ISSAA

Burton, Bruce Dr.
Intermetrics, Incorporated

Carlson, Susan Ms.
Ada Information Clearinghouse

Carlson, William Mr.
Intermetrics, Incorporated

Carrio, Miguel Mr.
MTM Software Engineering, Incorporated

Chantker, Arthur Mr.
Office of Naval Research

Colket, Currie Mr.
SPAWAR 331

Comer, Edward Mr.
Software Productivity Solutions

Cook, David Dr.
U.S. Air Force Academy

Cooper, Milton Mr.
Computer Sciences Corporation

Crafts, Ralph Mr.
Ada Software Alliance, Incorporated

Croak, Tom Maj, US Air Force
USAF/SCTA

Dancy, Bonnie Ms.
EVB Software Engineering

Davis, Carolyn Dr.
George Mason University

Depasquale, Gerry Maj, US Marine Corps
U.S. Marine Corps-MCCTA

Devlin, Michael Mr.
Rational

Dewar, Robert Dr.
New York University

Dickey, Michelle Ms.
Rational

Dikel, David Mr.
Applied Expertise, Incorporated

Dollahite, Bert LTC, US Army
ODISC4

Dowling, Debbie Ms.
Electronic Data Systems, Incorporated

Draham, Joseph Mr.
Electronic Data Systems, Incorporated

Druffel, Larry Dr.
Software Engineering Institute

Dumas, George Dr.
Directions Data, Incorporated

Dunbar, Terry Mr.
TLD Systems Limited

Dupaix, Leslie Mr.
USAF/STSC

Ehrlichman, Lee Mr.
Tartan, Incorporated

Elliott, Ronald Dr.
HQ USMC, C4I

Emery, David Mr.
The Mitre Corporation

Endoso, Joyce, Ms.
Government Computer News

Engle Jr., Charles Dr.
Florida Tech

Enright, Jim Mr.
U.S. Marine Corps-MCTSSA

Feldman, Michael Dr.
George Washington University

Fischer, Kurt Dr.
OASD (C3I), ODASD (IM)

Ford, Gary Mr.
Software Engineering Institute

Foreman, John Mr.
ARPA

Frakes, Bill Mr.
Virginia Tech

Gallagher, Jr., Edward Mr.
USACECOM SED

Garman, John Mr.
NASA Johnson Space Center

Glaser, Robert Mr.
Software Productivity Consoritum

Gogia, B.K. Mr.
Datamation System Research, Incorporated

Graham, Adele Ms.
ODISC4

Gregory,Donald Mr.
SAF/AQKS

Grieg, Joi Ms.
IBM Federal Systems

Gsellman, Larry Mr.
SETA Corporation

Hall, Sylvia Ms.
Ada Dual-Use Workshop Staff,
Electronic Data Systems, Incorporated

Hart, Hal Mr.
TRW

Hathorn, Fred Dr.
OASD (C3I)

Hemdal, Goran Mr.
RJO Enterprises

Henry, John Mr.
EDSI

Hess, James Mr.
DISC4

Hook, Audrey Ms.
Institute for Defense Analyses

Houlihan, Paul Mr.
DDC-I, Incorporated

Ada Dual-Use Workshop - Attendees

Howes, Norman Mr.
Institute for Defense Analyses

Joh, Jr., Frederick Mr.
Westinghouse Electric Corporation

Johnson, Barbara Ms.
Raytheon-MSD

Joiner, Harry Mr.
Telos Corporation

Kennedy, Gary Mr.
IBM Federal Systems Corporation

Kerner, Judy Ms.
The Aerospace Corporation

Kester, Rush Mr.
DC SIGAda

Kidwai, Shakil Mr.
Electronic Data Systems Incorporated

Kiviat, Philip Mr.
Knowledgeware, Incorporated

Kraft, Alan Mr.
Knowledgeware, Incorporated

Kramer, John Mr.
Institute for Defense Analyses

Krell, Bruce Dr.
Hughes

Langdon, Joan Dr.
Bowie State University

Langdon, Larry Mr.
Census Bureau, Commerce Department

Lee, Billy R.
Genesis Software Incorporated

Lee, Charlotte MAJ, US Army
Ada Joint Program Office

Leif, Robert Dr.
Ada-Med

Leong-Hong, Belkis Ms.
Defense Information Systems Agency

Leyden, Donovan Mr.
Defense Information Systems Agency

Lieberman, Wayne Mr.
Tartan, Incorporated

Long, Timothy Mr.
Ohio State University

Marciniak, John Mr.
CTA, Incorporated

Maresca, Paul Mr.
Adasoft, Incorporated

Martinez, Carlos Mr.
AETECH, Incorporated

Maymir-Ducharme, Fred Dr.
Unisys Government Systems

McAllister, David Mr.
Silicon Graphics Incorporated

McCall, James Mr.
SAIC

McGarity, M. Joan Ms.
NCTC

McKay, Charles Dr.
University Houston-Clearlake

McPherson, Mike Maj, US Air Force
SAF/AQKS

McPherson, Charles Mr.
U.S. Army Materiel Command

Miller, Philip Mr.
Carnegie Melon University

Milton, Donn Dr.
IIT Research Institute

Moran, Lindy CDR, US Navy
NCIC

Moskowitz, Paul Mr.
VERDIX Corporation

Moxley, Frederick Mr.
Defense Information Systems Agency

Munro, Neil Mr.
Defense News

Need, Perry Mr.
Ventura Technology

Needham, Donald LT, US Navy
U.S. Naval Academy

Nyberg, Karl Mr.
Grebyn Corporation

O'Kelley, Jeanne Ms.
ALSYS

O'Neill, Don Mr.
Independent Observer

Olie, Jean-Louis Mr.
ALSYS

Olsen, Al Mr.
Digital Equipment Corporation

Paige, Jr., Emmett The Honorable
Assistant Secretary of Defense, C3I

Petersen, Richard CAPT, US Navy
Defense Information Systems Agency

Plain, Russell Mr.
TRI-Pacific Consulting

Pleasant, James Mr.
Directions Data, Incorporated

Porter Wayne Mr.
Ada Dual-Use Workshop Staff

Potter, Marshall Mr.
NISMC

Quann, Eileen Ms.
FasTrak Training Incorporated

Reedy, Christopher Mr.
The Mitre Corporation

Reifer, Donald Mr.
Institute for Defense Analyses

Rhodes, Thomas, Mr.
NIST/CSL

Riehle, Richard Mr.
Adaworks

Ritter, Anne Ms.
Ada Dual-Use Workshop Staff
Electronic Data Systems, Incorporated

Roby, Clyde Mr.
Institute for Defense Analyses

Rogers, Dirk Maj, US Air Force
Ada Joint Program Office

Romano, Dan Maj, US Air Force
SAF/AQKS

Ross, Ron LTC, US Army
ODISC4

Rothrock, Jack Mr.
SofTech, Incorporated

Royce, Walker Mr.
TRW

Ruhf, Laura Ms.
Ada Dual-Use Workshop Staff,
Electronic Data Systems, Incoporated

Russomano, Gerard Mr.
Defense Information Systems Agency

Ryer, Mike Mr.
Intermetrics, Incorporated

Samadani, Hamid Mr.
SofTech, Incorporated

Sanden, Bo Dr.
George Mason University

Sapenter, Michael Mr.
Telos Corporation

Schwarm, Stephen Mr.
The Mitre Corporation

Schwenk, Robert Mr.
ODISC4

Short, Alonzo LTG, US Army
Defense Information Systems Agency

Shumskas, Anthony Mr.
LOGICON

Siegel, Barry Mr.
NCCOSC-NRAD

Sigrist, Bob Mr.
Electronic Data Systems Corporation

Sitaraman, Murali Prof
West Virginia University

Sodhi, Jag Mr.
U.S. Army, CECOM

Solomond, John Dr.
HQ Army Materiel Command

Spear, Jon MAJ, US Air Force
Naval Postgraduate School

Stevens, Scott Mr.
Magnavox Electronic Systems

Stewart, Howard Mr.
EG & G IDAHO

Straeter, Terry Dr.
GDE Systems Incorporated

Stuart, Antoinette Ms.
ASN (RDA)

Swartz, Leon Mr.
IBM Federal Systems Corporation

Tadda, George 1st Lt, US Air Force
U.S. Air Force

Taft, S. Tucker Mr.
Intermetrics, Incorporated

Thomas, William Mr.
Adirondack Computational Labs

Thomes, Col(RET), James Mr.
AETECH, Incorporated

Thurmond, II, George LTC, US Army
PEO STAMIS

Townsend, Eileen s.
Ada Dual-Use Workshop Staff,
Electronic Data Systems, Incorporated

Van Scoy, Frances Ms.
West Virginia University

Vasilescu, Eugene Mr.
Grumman Data Systems

Vicellio, Gary Mr.
The Mitre Corporation

Vorthmann, Scott Mr.
Carnegie Mellon University

Waterman, Deb
Statistica, Incorporated

Waters, Albert Mr.
U.S. Army, INSCOM

Wexelblat, Richard Dr.
Internal Revenue Service

Williams, Lyn Mr.
Bridge Enterprises, Incorporated

Willis, Robert Mr.
Hampton University

Wynn, Alden Mr.
Department of Energy, INEL

Zimmerman, Bill Mr.
Intermetrics, Incorporated

APPENDIX C

ADA DUAL-USE WORKSHOP
OCTOBER 19-20, 1993

ACADEMIA PANEL PROCEEDINGS

November 8, 1993

Defense Information Systems Agency
Joint Interoperability and Engineering Organization
Center for Information Management
Software Systems Engineering Directorate

TABLE OF CONTENTS

1.0 INTRODUCTION

This Appendix contains the findings, conclusions, and recommendations for the four questions that were posed to the academia panel. This data supports the summary of the academia panel as described in the Ada Dual-Use Workshop Summary.

2.0 ACADEMIA QUESTIONS AND FINDINGS

2.1 QUESTION #1: What actions can we take to make Ada one of the primary teaching languages within major colleges and universities?

FINDINGS

The following findings were a result of the participant briefings:

- Ada is a foundation language taught in the first or second course for computing instruction at approximately 100 institutions of higher education. Ada is taught somewhere in the curriculum of several hundred other institutions. However, this is a small "market share" of academia.

- In order to achieve greater market share as a teaching language for Computer Science (currently the largest computing discipline), Ada would have to supplant the predominant teaching language Pascal and Pascal's current academic challengers, C and C++. However, there is a real window of opportunity. Pascal is generally acknowledged as being antiquated, but C and C derivatives are perceived by the Pascal community as retreating to a pre-engineering past. Therefore, the Pascal community could be persuaded to embrace Ada as a logical software engineering successor (and a buttress against popular erosion of software engineering support).

- Computer Science instructors prefer to teach programming languages that they are also using in their research activities.

- Perceptions are a part of reality even when the perceptions are not true. Many educators still believe that Ada compilers are not widely available and are too expensive for educational institutions to adopt. This includes the belief that Ada compilers and tools will not execute satisfactorily on existing, personal computing platforms.

- Perceptions are a part of reality even when the perceptions are, but need not be, true. This includes the belief that demand is decreasing for Ada skills and knowledge and that U.S. funding agencies (i.e., ARPA and NSF) are anti-Ada.

- Some existing Ada 83 organizations and projects are doing some of the right things. We should intensify and build upon this work For example, GNAT, BAA, efforts by ASEET and the AdaIC are all committed to the advancement of the goals and objectives of Ada 9X.

- Ada is being short changed by the "Computer Science" focus on such issues as programming languages rather than on the more important issues of the software engineering life cycle.

- If research funds were available to do Ada-based research and curriculum development, academic interest in these subjects would increase.

- The DoD Ada community does not know how to effectively interact with the Ada academic community.

- The academic community is interested in the availability of Ada source code (including tools and environments coded in Ada) to bolster the position of Ada as a foundation language for disciplines of computing. This means free of charge and freely modifiable because source code is available.

CONCLUSIONS

- Ada, the programming language, is an important by-product and facilitator of software engineering. Promoting Ada 9X as another programming language will not be productive. Promoting Ada 9X in terms of its support for software engineering can contribute to achieving these goals.

- Ada is a superior teaching language because it provides the right constructs - including support for concurrency - for effective, disciplined development of reliable systems, and because the strong standard and compiler validation makes Ada programs much more portable than programs in any other available language.

- In spite of the above advantages, "major universities" are characterized by their research. Until there is sustained and appropriate funding for Ada-related and Ada-based research initiatives, faculty at these (and other) universities will do their research (and teaching) in newer or what is perceived-to-be more fundable languages. Other colleges and universities frequently follow the lead of these "majors."

- The GNU-NYU Ada Translator (GNAT) will serve a uniquely effective purpose in bringing Ada 9X to academic researchers. A very high proportion of academic institutions already have GNU software installed. GNAT will be distributed as an inherent part of the next version. This means that no major university will be without a usable Ada 9X compiler. the availability of source code for this compiler will enable GNAT to become an important vehicle for academic research projects. This will improve Ada's acceptance in research and cause a "trickle-down" effect to computing curricula.

2.2 QUESTION #2: What initiatives can we mount to increase Ada's appeal, adoption and popularity within the academic community?

FINDINGS

The following findings were the result of participant briefings:

- The Ada community is isolated and the Ada programming language has not been properly promoted. Ada needs to be marketed more broadly to increase Ada awareness within the broader academic community.

- The AdaIC does good work popularizing Ada, but is inadequately and inconsistently funded.

- Computer science based research efforts are not adequate for a wide variety of efforts that could make Ada more popular at academic institutions (e.g., distributed systems, high integrity systems, MIS).

- Ada's lack of bindings to popular packages such as x-windows, data bases and spreadsheets impedes its broader use within academia.

- Curriculum development workshop, Ada training workshops and supporting software would all stimulate greater Ada awareness and appeal in academic institutions. Travel and per diem money is a necessity to assure good academic attendance at such events.

- Ada libraries with useful software for a variety of domains and accessible to the academic community are desirable by academia. However, many of the DoD reuse libraries are inaccessible to educators and have arcane user interfaces that reflect only source code concerns.

- Academic institutions could participate in designing and performing experiments showing the benefits of Ada 9X.

- Ada 9X could be positioned as a powerful tool for object-oriented programming and component-based programming. These capabilities could be researched in colleges and universities to increase academic community awareness.

CONCLUSIONS

- Even without a coordinated, full-fledged partnership of government, industry, vendors and academia, the participants in evolving and applying Ada 83 have made notable progress toward goals complimentary to those of this workshop. The infrastructure and accomplishments of this past work should be leveraged as a foundation for subsequent improvements.

- The goals, rationale and successes of Ada 83, both within and outside of DoD, have not been widely disseminated and marketed. Ada 83 is a rich and successful language and has little to apologize for. Ada 9X will be even richer and more successful. The Ada-advocates within academia have had scarce resources and incentives to help promote the needed outreach.

- Ada should be the foundation language (i.e., the language taught to majors in as many community colleges, colleges, and universities as possible). Ada should be the first, but not the only language taught in computing curricula Text books and other supporting instructional resources should be developed to use Ada as the language of choice in advanced courses such as operating systems and data base management systems.

- A comprehensive, integrated, software engineering curriculum should be evolved. The SEI, along with organizations such as ASEET, should play a leading role in this evolution.

2.3 QUESTION #3: What investments can we make to increase the commercial use of Ada?

FINDINGS

The following findings were the result of participant briefings:

- Ada 9X compilers and related tools could be used to stimulate interest in Ada.

- Lack of Ada bindings to popular windowing and database packages impedes the use of Ada in commercial applications.

- The academic community thought that advertising could be a key area in attracting interest to new products and technologies such as Ada.

- Ada needs more investment in CASE tools and products targeted for personal computers.

- Ada is often perceived as applicable for only large scale projects in the defense industry or for embedded systems in general. These misconceptions should be cleared up through a public relations program.

CONCLUSIONS

- Although it is easy and popular to recommend increased marketing and public relations activities as the key to commercialization of Ada 9X, such activities are only a subset of the broader and more important issues of successful technology transfer.

- Berkley Unix and X-Windows (trademarks) are only two examples of academic research products that have had a major impact on commercial software products. With planning, leadership and funding, Ada products of academic research projects can be used to address next-generation computing problems with similar results.

2.4 QUESTION #4: What steps can we take to increase the pool of entry level personnel with Ada skills, knowledge, and experience?

FINDINGS

The following findings were the result of participant briefings:

- Many of the findings to question #1-3 are applicable to this question.
- Student internship programs were cited many times as a way to develop "hands-on" Ada skills during college.
- Ada instruction needs to be expanded beyond a single course to a set of courses emphasizing software engineering that begin in high school and continue throughout college.
- The software engineering community draws practitioners from numerous disciplines. Efforts to increase the number of Ada skilled personnel should not be limited to the computer science field.
- There is a perception that DoD and ARPA over-emphasizes the influence of major research institutions in training Ada programmers and researchers. A successful dual-use program has to build on the important and sizable grass-roots constituency in hundreds of colleges and universities throughout the country. DoD needs to learn how to communicate with and to mobilize that constituency. This is a successful and enthusiastic base of educators to build upon.

CONCLUSIONS

- Government procurements have failed to incentivize industry and vendors to emphasize Ada education, training, experience, achievement, and excellence. The result is that academia is concerned with the lack of jobs available for graduates that want to work in Ada-related projects.
- Satisfying the issues of questions #1-3 would help resolve the issues of questions 4 and vice versa.

3.0 OVERALL RECOMMENDATIONS

The following recommendations were derived from the findings and conclusions listed above:

- This two day workshop produced a useful discussion of key issues and possible approaches for institutionalizing Ada 9X within academia and thereby contributing to the commercialization of 9X. However, continuing dialogue and cooperation will be required if academia is to be involved as a full-fledged partner with government, industry, and vendors in achieving the goals identified for this workshop.

- The following recommendations require a very small amount of additional DoD investment compared to the potential benefits and to the potential costs of failing to make such an investment.

1. Low cost/free environments, tools and other resources supporting instruction and research in and with Ada 9X should be made available to educators and students as soon as possible.

   Discussion: Such support should be consistent with the software engineering goals and objectives of 9X and rewarding for the initial and the more experienced user. At the entry level, the visual interfaces and other support for student learning must be at least as good as that now provided in competing languages and environments such as C++ and Turbo Pascal (trademarks). In addition, the Ada 9X environments, tools, and resources must clearly support the principles of software engineering.

2. Expand an existing forum, Ada Software Engineering Education and Training (ASEET), to serve the needs for timely information and technology exchange among all educators that are stakeholders in Ada 9X.

   Discussion: Membership should include, but not be limited to, educators from: military academies, universities, colleges, technical institutes, community colleges, high schools, commercial training firms and Software Engineering/Ada instructors within an organization.

3. Libraries of reusable components useful for instruction and research should be made available at little-or-no cost to educators.

   Discussion: Such components should span the software engineering life cycle and include bindings, interfaces and implementation components.

4. DoD should sponsor an initiative to effect curriculum improvement that covers and integrates issues across the software engineering life cycle (Preferably, this should involve a number of other organizations as cosponsors).

Discussion: The initiative should include the use of Ada 9X in a variety of domains including, but not limited to, mission critical embedded systems, C3I and MIS. Such an initiative should build upon other efforts such as the ARPA BAA and the AJPO TIP. However, this initiative should be focused upon software engineering as a whole and upon relevance to education and training in particular. In addition to ASEET (see Recommendation 2), the SEI should serve as a focal point for this work.

5. NSF-style faculty development workshops should be made available to facilitate the expansion and integration of Software Engineering and Ada 9X instruction and research in academia.

   Discussion: This means travel, per diem, and a small stipend should be available for educators who are interested in participating in such workshops.

6. An Ada-based, software engineering, research agenda for at least the next five years should be established and publicized.

   Discussion: Such an agenda should include the kind of large-scale, basic research issues now addressed by ARPA initiatives. At the other end of the spectrum, the agenda should also include opportunities for smaller-scale, applied research projects that might be addressed by new faculty members in smaller institutions. This means, for example, that a new faculty member interested in Software Engineering and Ada 9X could depend upon the availability of related research funding for a sufficient period (i.e., at least five years) to establish an academic career.

7. Fund instruction, research, and service that helps place Software Engineering and Ada 9X in "front of the wave."

   Discussion: Strengths of Ada 9X, such as reliability and maintainability, are not as attractive to potential students as perceived applicability to such current hot topics as virtual reality. However, virtual reality is only one of the current hot topics that could benefit from Ada 9X. sponsorship of Ada 9X based research in such subjects along with sponsorship of student team competitions can raise the visibility and desirability of including Ada 9X and Software Engineering in academia.

8. Restore/initiate consistency in DoD's treatment of Ada and Software Engineering.

   Discussion: Say what you mean. Mean what you say. Budget and act accordingly. The supporters of Ada and Software Engineering within academia have paid a very high price for the vacillations anc inconsistent signals of DoD. A mandate that is ignored without consequence, that is unsupported by any major, Ada-based research initiatives in academia, and that is not generally perceived to relate to anything other than the issues of DoD's programming language preferences undermines the best interests of all stakeholders and strongly argues against the inclusion of Ada in academia.

9. An outreach program should be mounted in conjunction with implementing the above recommendations so that the necessary public relations, marketing and other technology transfer activities support the timely insertion of Software Engineering and Ada 9X into education and training.

   Discussion: The AdaIC should play an important role in this dissemination.

10. Coordinate a long term, cooperative program among government, industry, vendors, and academia to advance the goals and objectives for Software Engineering and Ada 9X.

    Discussion: Sponsorship of thesis students, faculty and student researchers, cooperative research and instruction, internships and exchange programs are some examples of such coordination.

4.0 BRIEFING SUMMATION

This section lists the major issues and observations provided by the participants in the workshop. Each participant was allowed 15 minutes to address the four questions. The panel members then evaluated each issue and observation, to produce the consensus recommendations in the main body of the report. Not all of the issues and observations listed in this appendix were considered valid or significant by the panel, but all of the participants added considerable value by raising ideas and stimulating discussion.

4.1 SCOPE

Major research universities, teaching colleges, and two-year institutions should all be supported by any efforts to increase the use of Ada in academia. The smaller universities and the community colleges produce far more entry-level software engineers than the research schools, and getting Ada into those institutions is at least as important as getting it into institutions like MIT and CMU. The panel's observations and recommendations are addressed to all types of two and four-year, post-high school educational institutions. The commercial training industry is an important supplement, but this panel did not develop any specific recommendations to aid it. One participant suggested that an effort to increase Ada teaching in high school was needed, but this received only limited support.

4.2 THE FOCUS OF ADA EDUCATION

Software Engineering was described as the main focus of Ada education efforts by many participants. It was pointed out that "Engineering" is a word that is "owned" by the Engineering (not Computer Science) part of many Universities, and that in some states Engineers of all types have to be certified and registered, so this term is not readily usable at all institutions. The consensus was that Software Engineering, as we have traditionally understood it, is the right focus, but that another term may be needed.

Another observation was that making new graduates *comfortable* with Ada is at least as important as making them fluent in Ada. When they reach industry, students will resist languages they are not comfortable with. If they feel that Ada is a sensible, learnable, useful, modern language, they will not resist it even if they are not used it themselves. Ada should not be the only language taught. One participant stated that a "computer scientist" who knew only one language would be unemployable.

Another common theme was that Ada should be taught as a means, not as an end. For example, it is very effective to teach a course on artificial life, using Ada as the implementation language. An exciting main topic motivates students and makes learning Ada easier.

It was also suggested that Ada Education efforts should focus on "component engineering".

4.3 ENSURING A MARKET FOR ADA PEOPLE

If the majority of employment advertisements for software professionals named Ada as a mandatory or preferred language, Ada would immediately become the foundation language in most universities with no further action by the DoD.

There was general agreement that there are already enough Ada-qualified people for the jobs that require the use of Ada. More people knowing Ada would tend to increase the number of projects that use Ada by choice, however. Though a significant number of large DoD programs are using Ada, very few DoD facilities use Ada for the majority of the work done in house or by local contractors. For example, one participant had recently surveyed DISA's internal requirements, and determined that C and C++ skills were needed much more than Ada skills for current work. Thus, Ada is not a strong requirement in the job market even in locations which are dominated by a DoD laboratory.

One participant stated that advertisements may stress C or C++ more than the employers really mean, because naming particular programming languages and systems is easier than identifying software engineering skills in the communication from engineering through personnel departments and into a few square inches of newsprint. Nonetheless, market demand is a critical factor. There was a consensus that anything the DoD can do to increase its own, or private industry's, use of Ada will also expand Ada's use in academia.

4.4 RESOURCES

The use of Ada as a foundation language (the first language taught to scientists and engineers) is dependent on resources. Currently, there are more and better resources to support other languages, as discussed below.

4.5 CURRICULUM DEVELOPMENT

One of the most common themes was that a full, multi-year, Ada-based Computer Science and/or Software Engineering curriculum, with associated books, materials, and exercises, was needed. Using Ada continuously, from entry level courses through Junior and Senior year courses and projects was seen as very important.

There are two external impediments: The AP (advanced placement) test requires Pascal; and the AACSB (American Association of Colleges and Schools of Business) includes COBOL as part of its mandatory curriculum. Changing either requirement will take a concerted effort over many years, and must be in the context of a growing Ada market share.

The DARPA BAA for curriculum development was considered extremely valuable. Several participants suggested that it be better funded and better advertised. It was also recommended that two current rules be relaxed: participants wanted the BAA to cover the updating of existing course material to Ada to be covered instead of only new courses, and they wanted multi-year projects to be allowed. There was a consensus that it takes several years to develop a good curriculum and its supporting materials.

It was stated that running the BAA through DARPA lends prestige. It was also stated that DARPA-run procurements are not accessible to smaller universities, and other schools; the smaller institutions expect to get most of their money from the National Science Foundation. The NSF is also farther removed from the military, which makes it a better funding agency for many faculty members.

There are many textbooks on Ada for students. However, there is no textbook for the teachers. We need a "how to teach Ada" manual to make it easier for new adopters.

There is also a shortage of textbooks that *apply* Ada. It would be very helpful if the language used in texts on AI or operating systems, for example, used Ada for its code samples. Some textbooks come with a floppy disk containing C source code, but none come with Ada.

4.6 PROOF THAT IT WORKS

One participant said "we must stop relying on anecdotes and opinions and provide real data on the effectiveness of Ada in education". A large-scale, hypothesis/experiment scientific study was recommended. The group recognized the value in this, but their implementation details were not worked out. One participant recommended post-graduation interviews with students, to learn whether they are applying software engineering and/or Ada skills in their jobs.

4.7 FUNDING RESEARCH

Most people defined "major schools" as the leading research colleges. The non-major schools turn out far more graduates, and their use of Ada is also very important, but they do follow the lead of the research colleges to a significant extent.

NSF's selection is based on Peer Review. Regardless of top management intentions, basing an NSF research proposal on Ada practically ensures that the proposal will be rejected during peer review. It was suggested that some bucket of funding be set aside at NSF for University research related to Ada. Everyone agreed that it would be unproductive for DoD or Congress to pressure NSF to require Ada, or even to favor it. NSF likes to do joint programs, and if DoD can provide part of the funds for projects, it can specify Ada with no problem.

Another suggestion was to seek out leading Computer Science researchers and offer them an opportunity and funding to connect their work to Ada. Supporting research requires appropriate reusable components. GNAT is an excellent start. AdaSAGE would also be useful, especially in for Information Systems students, if it were more available to universities. There was widespread sentiment that this government-funded product should be available with full source code at no cost.

There is also a need for implementations of operating systems and other high-tech software systems in Ada. Bindings to systems in other languages serve the needs of the production community, but for research to be connected to Ada it is necessary that the *implementation* be in Ada as well.

Some participants recommended that *every* DoD grant and contract have some Ada connection. However, others said that this would be unacceptable to the research community. Providing "extra points" in evaluation for Ada related programs was broadly supported, however.

One way to increase the use of Ada in research would be to provide assistance in writing grants to faculty members that propose Ada. The DoD could provide a staff which helped to find opportunities and/or to create the actual application or proposal.

Senior researchers working in C or C++ put pressure on the entry-level instructors to teach C and C++ so the students are ready to join the project in the later years.

4.8 TOOLS AND ENVIRONMENTS

"*We must give Ada new adv*antages with respect to tools and software components".

Universities choose languages more on the associated tools and environments than on the features of the language itself. For education, tools need to be "student friendly". Aspects of this that were stressed by the participants were:

- **A Gradual Learning Curve**: Students should get some results immediately, rather than having to learn a whole system before they can use it. The paradigm is "learn a little, do a little".

- **Fun**: Students need to be interested, challenged, and excited. Tools and environments for students should use high-quality GUIs, multi-media, graphical views of programs, and generally attractive technology.

- **Moderate Resources**: Most colleges have PCs, and not the most modern and powerful PCs at that. While the community appreciates the Ada vendors efforts to make their products available for free or minimal cost, it does not meet the need when an Ada environment requires high-end workstations with large amounts of RAM for reasonable performance.

- **Avoid Blocking on Syntax**: Students need to be able to concentrate on the impoꞏtant issues. Student tools should help with syntax so the students can learn real computer science rather than where to put the semicolons.

Minimal requirements include: multi-window editors, full compiler integration, graphical display of dependencies, automatic recompilation, and a navigable library manager.
Student-fast means that there is no time for a distraction to set in while waiting for a compilation.

Today's Ada tools do not generally meet the ease-of-use standard set by current C and C++ tools. Novices require highly visual environments.

There was a call for a "free Turbo-Ada for Universities". Objections were raised that this could hurt the commercial compiler industry, and that current prices were low enough

"It's not enough to teach Ada and generate enthusiasm if in industry the Ada tools and environments don't work".

There are many useful tools out there; what's needed is to add the missing 10-20 percent, and ensure that these systems actually reach the educators who need them.
There was a wish that Microsoft and Borland would sell Ada compilers and environments. It was acknowledged that this would be extremely difficult to accomplish until such time as Ada was a bigger market than C or C++.

4.9 GETTING IN FRONT OF THE WAVE

The "essentially cool stuff" that computer science students want to use and learn about is being written in C, C++, LISP, and other languages. Virtual reality, AI, parallel processing, multimedia, and other state-of-the-art, exciting technologies are the interests of students, and the proper focus of computer science departments. The experimental and prototyping work researchers do under DoD funding is widely considered to be "outside" the DoD Ada mandate. The DoD can always redevelop production versions of this software in Ada, or write an Ada binding to it later, when the technology is sufficiently mature for use in weapon or information systems, but this approach does not address academia's needs. It leaves Ada always behind the wave, catching up with technology that was first available in C, LISP, or some other language.

Engineering students (whether Software Engineering or another field such as Electrical Engineering) are less sensitive to the implementation language used in computer science research. However, Ada's current position behind a another wave does limit its appropriateness for these students: this is the tools and environments wave discussed above.

To rapidly expand the use of Ada as a foundation language in software engineering and computer science education, it is necessary to get ahead of the wave in research artifacts and in development environments. This could be done by funding a large-scale virtual reality research project that would use Ada, for example. A development environment better than Visual Basic or TurboC is needed to get ahead of the tools and environments wave. Ada environments must become technically better than Borland's C++.

DoD's repositories are not serving academia's needs. They are not accessible without burdensome paperwork and delays. Though access may be "streamlined" from the viewpoint of DoD contractors and project offices, the academic community compares it with the instantaneous, paper-free process it now uses (anonymous ftp over the internet, coupled with informal archiving/indexing systems maintained on a volunteer basis). Nearly all four-year institutions have internet access, at least for selected faculty, and can tap into hundreds of archives, containing millions of lines of useful, working code. Cross-repository searching mechanisms are informal, but can be used very effectively by experts. All of this software is either public domain or otherwise usable for university research without permission, paperwork, fees, or delays. Very little of this software is in Ada. In comparison to the C and LISP software instantly available on the Internet, DoD's Ada software is locked up and unindexed. DISA is not connected to the Internet.

4.10 OUTREACH

Steps are needed to improve Ada's image in industry and academia, and to put Ada educational resources into the hands of educators, as discussed below.

4.11 PUBLIC RELATIONS

The name of the AJPO sounds too military. It should be changed. Anything would be an improvement. Call it "snoopy" or "AdaSEX". Avoid the military image, which is very undesirable to students and faculty. "I am *not* going to teach a language that is primarily for building bombs". We should never use military or violent example programs in instruction and publicity. If industry gives tools to universities, they should come back later and find out how its working out. We should make a disk (with a picture of Ada on the cover) containing success stories, summaries of resources, statement of commitment, and general advertising for Ada; make hundreds of copies; and give them away free at conferences (not TriAda, *outside the Ada community*, and targeted to the education community. That we must stop preaching to the choir was a common theme. We should participate in SIGPlan as well as SIGAda.

A consistent message from the DoD is an essential part of any public relations effort. The periodic anti-Ada blasts from high DoD levels are very harmful. The DoD needs to *consistently* support Ada in both words and actions.

Pascal is the foundation language in nearly half of current University programs. It is being gradually displaced by C or C++. Ada could and should go after the Pascal share of the market. The C language rode in on the coattails of UNIX. Ada needs something that does the same thing UNIX did for C. Reuse, reliability, portability, maintainability and the other major features that make Ada attractive to the DoD are not attractive to students. Students want to learn and use languages that are modern, interesting, popular, fun, and "essentially cool". We shouldn't allow Ada to be compared with non-Software Engineering languages.

4.12 HELPING FACULTY GET STARTED

Educators are busy. They tend to teach the same course year after year, with the same materials. They would be happy to update the courses, but don't have time. This inertia handicaps Ada introduction. People teach what they know, like, and use.

The DoD should support a touring team of Ada educators to help universities get started teaching and using Ada. The ASEET program is a good beginning, and very modest additional travel funding coupled with volunteer efforts from academia will help. The purpose of these teams would not be to persuade universities to switch to Ada: it would be to help them get started once they have chosen Ada. However, knowing that the help was available would be a strong positive for Ada in their decision. The DoD should fund summer workshops to teach Ada and Ada instruction to educators. The NSF operates such workshops in many other technology areas, and this would make a good model. Educators should receive travel and lodging costs and perhaps a small stipend to attend.

4.13 FURTHER OBSERVATIONS

Though not recommendations or issues, the following observations were made and well-received by the panel:

- All faculty members are eager to do research and get funding for it.

- Stars and the SEI get lots of money, but haven't produced many products useful to the academic community.

- Even educators who currently use C and C++ as a foundation languages acknowledge that they are poor languages from a teaching viewpoint. C or C++ are chosen because they are *popular* and well supported by tools and other resources. Ada is clearly better as a language, but more resources and more "outreach" are needed to make it a practical choice for many institutions.

- There is enormous "downward pressure" on introductory-level instructors to switch to C. This is because industry and students believe that they can get jobs with C skills, but not with Ada. This perception (which may be a fact) has a strong influence on University management.

- "Ada Research" is not respectable. It is like doing "COBOL research". Ada can be used in many other research projects (some people argued *all* other research), but is not a topic in itself.

- The DoD never had a reasonable plan to incrementally transition its own staff and work to Ada. DoD had the intention, but never created a real, tactical plan. A "big bang" transition could never have worked.

- To make Ada succeed, 90% of DoD people (military, civilian, *and transients such as DARPA managers*) must get on-board.

- Academics are beginning to realize how bad C is (for software engineering and teaching in general), and that C++ doesn't fix it. Beginners can do too many things wrong, and the compilers don't help. Ada slaps your hand if you write something stupid; C assumes you know what you're doing.

- Summer funding is extremely valuable in small colleges, because salaries and supplemental funding for instructors are tight in their existing budgets.

4.14 OTHER RECOMMENDATIONS FOR DOD ACTION

- "The DoD should never buy a piece of hardware without an Ada compiler on it". DoD needs to actually buy the compilers with the hardware, not just note that one is available. Systems which don't have *any* compilers or development tools don't need Ada compilers either, but if the machine supports development in any other language, it must have support for Ada as well. This point is tied to the need for a consistent message from DoD and the need to stimulate the compiler and tools industry to make better products.

- In commercializing Ada, the DoD should target particular application domains, or market niches, and invest in the infrastructure for each niche. Examples of domains are Simulation, On-Line Transaction Processing, and Computational Fluid Dynamics. In each targeted domain, it would be necessary to develop Ada equivalents of the important reusable software and COTS tools now used in that domain.

- Given the necessary infrastructure, the DoD could sponsor pilot projects or "shadow" programs to show how effective Ada is in that domain.

- The DoD should require that all COTS software it purchases provide equal support for Ada: If the product provides a programming interface to C, COBOL, or any other language, it must also provide one for Ada. It would not be necessary that the COTS products be written in Ada, only Ada bindings be included for any programming interfaces.

- If the DoD would incentivize software engineering and software reuse, and include life-cycle costs in the evaluation of program managers, it would increase the use of Ada on DoD programs.

- A minority held that the mandate was harmful because of "reverse psychology", but the majority supported the mandate because of the needs for an assured base of employment opportunities for Ada-trained graduates and for a consistent public Ada message from DoD.

- Providing companies who work in Ada with small financial incentives to bring in co-op or summer students would have great leverage: it would create a demand for Ada-fluent students in the universities, give the students experience and success stories, and encourage more contractors to use Ada.

5.0 PANEL

Panel members were selected to represent various facets of the industrial community.

	Name	Organization
Panel Chair:	Dr. Charles McKay	University of Houston - Clearlake
Panel Facilitators:	Lt Col Dennis Bowers	Defense Information Systems Agency
	Mr. Michael Ryer	Intermetrics, Incorporated
Panel Members:	CAPT David Cook	US Air Force Academy
	Dr. Michael Feldman	George Washington University
	Dr. Gary Ford	Software Engineering Institute
	Ms. Eileen Quann	FasTrak Training
	Dr. Francis Van Scoy	West Virginia University

6.0 PARTICIPANTS

Participants were allocated a maximum of 15 minutes to present the panel members their comments to questions within the academia discipline (See questions listed in the Ada Dual-Use Workshop Academia Summary, section 6.0 of this report). The following participants presented briefings during the Ada Dual-Use Workshop (Refer to Section 7.0 of this Appendix, Participant Briefings for copies of presentations). Participants are listed in the order of presentation.

Name	Organization	Briefing
Sitaraman, Murali	West Virginia University	Presentation
Plain, Russell	Tri-Pacific	Presentation-Slides
Bedar, George	U.S. Naval Academy	Presentation
Vorthmann, Scott	Carnegie Mellon University	Presentation-Slides
Dewar, Robert	New York University	Presentation-Slides
Moran, Lindy	NCIC	Presentation-Slides
Bolz, Richard	Ventura Technology	Presentation-Slides/Position Paper
Potter, Marshall	NISMC	Presentation-Slides
Frakes, Bill	Virginia Tech	Presentation
Vasilescu, Eugen	Grumman Data Systems	Presentation-Slides
Long, Timothy	Ohio State University	Presentation
Miller, Philip	Carnegie Mellon University	Presentation
Engle, Charles, Jr.	Florida Tech	Presentation
Kramer, John	Institute for Defense Analyses	Presentation
Spear, Jon	Naval Postgraduate School	Presentation-Slides
Needham, Donald	U.S. Naval Academy	Presentation
Willis, Robert	Hampton University	Presentation
Adams, Elizabeth	Hood University	Presentation
Langdon, Joan	Bowie State University	Presentation

Note: Some of the views expressed were solely the participant's own and not of his/her organization

7.0 PARTICIPANT BRIEFINGS

This section contains a copy of participant briefing materials provided to the Industry Panel during the Ada Dual-Use Workshop. In cases were presentation materials were submitted in handwritten form, a typed copy was created by the Ada Dual-Use Workshop staff and are noted accordingly.

Participant Briefing
Murali Sitaraman
West Virginia University

Software Component Engineering and the Future of Ada

Murali Sitaraman
Dept. of Statistics and Computer Science
West Virginia University
Morgantown, WV 26506 - 6330

Correspondence: murali@cs.wvu.edu

Key Points

- Emphasize the influnces of introducing software component engineering principles early and throughout the CS curriculum on the US industry
- Fund design of multi-year course sequences to introduce SE principles
- Fund long-term experimental evaluation

-------------------- Bottom line --------------------

Hypothesis:— Strengths of Ada will make it the ideal choice for a software component-based curriculum

Note: The Curriculum Development initiative is right on the money, but does not currently support long-term investigations.

Why SE Principles Should Be Introduced Early?
Or
Problems Facing Undergraduate CS Education

- Absence of a context to learn fundamental principles such (e.g., abstraction) early in the curriculum
- Absence of motivation for (formal) specification and design issues early in the curriculum
- Relatively late exposure to SE principles resulting in graduates with little experience in practising SE
- Minimal exposure of SE principles for non-computer science majors; and
- Software construction not viewed as an assembly of "well-engineered" reusable components.

An Example Sequence

CS 1 (introductory course)
CS 2 (data structures course)
CS ? (algorithms analysis)
CS ? (compilers)
CS ? (software engineering)
CS ? (component engineering)

An Example Issue

Specification-based generic component reuse across courses — facilitated readily by Ada

Long-Term Evaluation

- Evaluate the influences of teaching SE principles on student careers in academia and the industry
- Evaluate abilities of multiple programming languages in learning SE principles
- Evaluate the influences of interleaving languages
- Evaluate the influences at a spectrum of institutions

Results will be two-fold

- Experimental evidence that Ada is an excellent vehicle for introducing SE principles
- Ada and serious Ada-competitors (e.g., C++) can co-exist in academia

Actions for Making Ada A Primary Language and Initiatives for Enhancing Ada's Appeal

1. Stress the importance of integrating software component engineering principles throughout the curriculum; evaluate influences on student careers

2. Support multi-year curriculum redesigns to instill software engineering/software reuse principles

3. Make a coherent series of textbooks available for such SE-based course sequences

4. Show experimental results on comparing/interleaving Ada with other programming languages

Investments

Fund 1, 2, 3, and 4 in the previous slide:

Fund multi-year, multi-institutional efforts for integrating software component engineering principles in CS curriculum and evaluating long-term influences of such efforts

Participant Briefing
Russell Plain
Tri-Pacific Consultants

Tri-Pacific Consulting Corp
1070 Marina Village Pkwy
Alameda, California 94501
510 814-1770
FAX 510 814-1768

Ada Dual-Use Workshop

Russell Plain
Tri-Pacific Consulting Corporation
1070 Marina Village Parkway, Suite 202
Alameda, CA 94501

Tri-Pacific Consulting Corp
1070 Marina Village Pkwy
Alameda, California 94501
510 814-1770
FAX 510 814-1768

Ada Dual-Use Workshop

Background

What actions can we take to make Ada one of the primary teaching languages?

What initiatives can we mount to increase Ada's appeal, adoption and popularity?

What investments can we make to increase the commercial use of Ada?

What steps can we take to increase the pool of entry level personnel with Ada skills?

Tri-Pacific Consulting Corp
1070 Marina Village Pkwy
Alameda, California 94501
510 814 1770
FAX 510 814 1784

Background

Our mission: technology transfer including training, mentoring, project support

Our experience:

- Avionics, simulation, communications, process control, MIS
- Databases, user interfaces, OSs, networks
- Embedded systems, application systems
- DOD, DOD subcontractors, non-DOD

Tri-Pacific Consulting Corp
1070 Marina Village Pkwy
Alameda, California 94501
510 814-1770
FAX 510 814-1784

Background - Things That Work

- Varying class length
- Emphasis on software engineering
- Domain specific
- Varying emphasis on different language parts
- Emphasis on labs

Tri Pacific Consulting Corp
1070 Marina Village Pkwy
Alameda, California 94501
510 814-1770
FAX 510 814-1788

Background - Things That Do Not Work

- Fixed class length
- "One size fits all"

  Entire language

  Same emphasis on all language constructs
- Unfriendly Ada environments
- Lack of relevance to a domain

Tri-Pacific Consulting Corp
1070 Marina Village Pkwy
Alameda, California 94501
510 814-1770
FAX 510 814-1788

What actions can we take to make Ada one of the primary teaching languages?

- Ada is not just a language, it represents:

  Software engineering

  A mindset change

  Emphasis on maintenance not development
- Undergraduate software engineering degree with Ada at its core (recommended at ASEET 1989)
- Attract "Engineers"
- Stop giving Ada lip service

  Make it a significant part of the Computer Science curriculum

  Give it more exposure than a "sampler" in a multi-language course

  Do not compare it to other languages without software engineering
- Tell students they will get jobs if they learn it

Tri-Pacific Consulting Corp
1070 Marina Village Pkwy
Alameda, California 94501
510 814-1770
FAX 510 814-1784

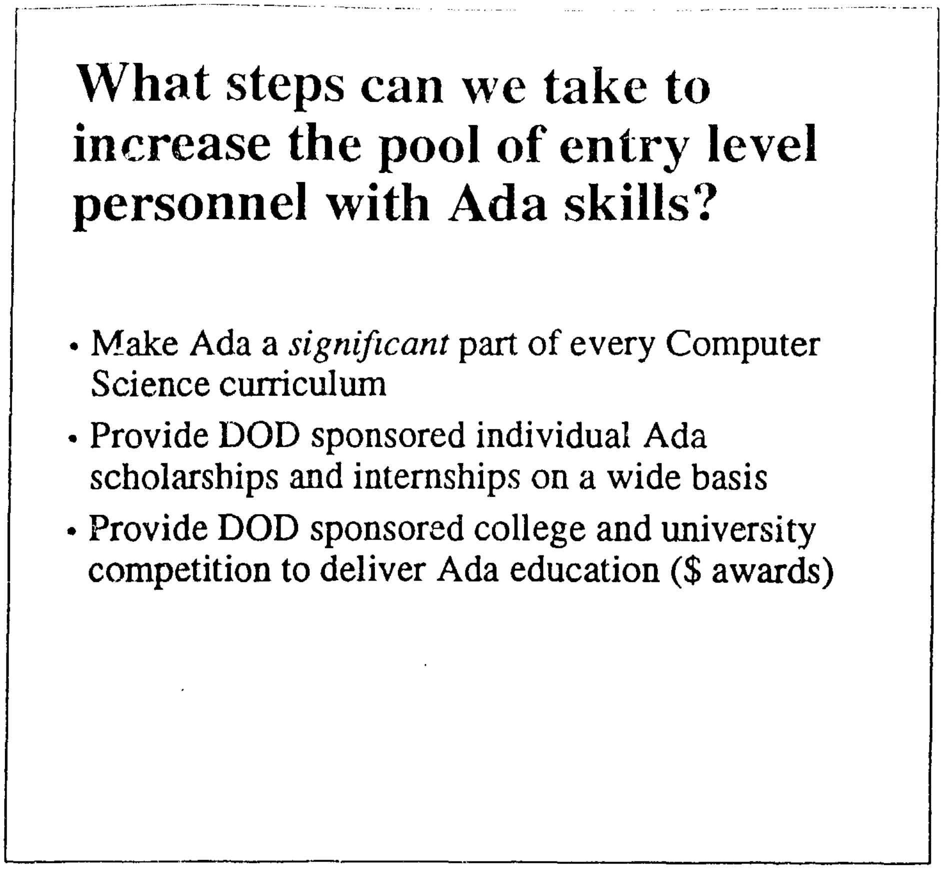
What initiatives can we mount to increase Ada's appeal, adoption, and popularity?

- Provide incentives for change
- Emphasize Ada *before* other languages
- Provide inexpensive compilers to students ($49)
- Emphasize software engineering *with* Ada
- Promote Ada 9X as a *future* language
- Provide a consistent (top to bottom) DOD Ada message

Tri-Pacific Consulting Corp
1070 Marina Village Pkwy
Alameda, California 94501
510 814-1770
FAX 510 814-1784

What investments can we make to increase the commercial use of Ada?

- *Give* schools compilers / environments
- Develop and implement software engineering with Ada *awareness and enlightenment* programs
- Promote a *positive* image with computer, academic and public press (not negative image or no image)
- Provide better access to Ada repositories



What steps can we take to increase the pool of entry level personnel with Ada skills?

- Make Ada a *significant* part of every Computer Science curriculum
- Provide DOD sponsored individual Ada scholarships and internships on a wide basis
- Provide DOD sponsored college and university competition to deliver Ada education ($ awards)

Participant Briefing

Scott Vorthmann
Carnegie Mellon University

(Presentation materials were submitted in handwritten form, a typed copy was created by the Ada Dual-Use Workshop staff.)

Scott Vorthmann

School of Computer Science
Carnegie Mellon University

vorth@cmu.edu

Premise:

The advantages of C are largely due to history: the quantity and diversity of software and tools available have made language features secondary in language adoption decisions.

Course of action:

We must give Ada new advantages with respect to tools and reusable software:

- Tools with new functionality, support for interoperability, prototyping, etc.

- Highly reusable software libraries, including Ada implementations (not just bindings) of OSs, DBs, and GUIs

- MUST VIEW PROMOTION OF Ada IN A CONTEXT OF COMPETITION WITH C, C++
- MUST PROVIDE INCENTIVES THAT "LEAPFROG" CAPABILITIES OF EXISTING C LIBRARIES AND TOOLS
- MUST IDENTIFY NEW TECHNOLOGIES IN THIS REGARD, AND SUPPORT THEM WITH Ada PRODUCTS
- NEED TO FIND THE "UNIX" OF Ada - ONE OR MORE VEHICLES, WRITTEN IN Ada, THAT MAKE THE LANGUAGE HIGHLY ATTRACTIVE
- Ada BINDINGS ARE IMPORTANT, BUT INSUFFICIENT; IMPLEMENTATIONS OF REUSABLE LIBRARIES, ETC. ARE IMPORTANT FOR ATTRACTIVENESS IN RESEARCH AND COMMERICAL VENUES

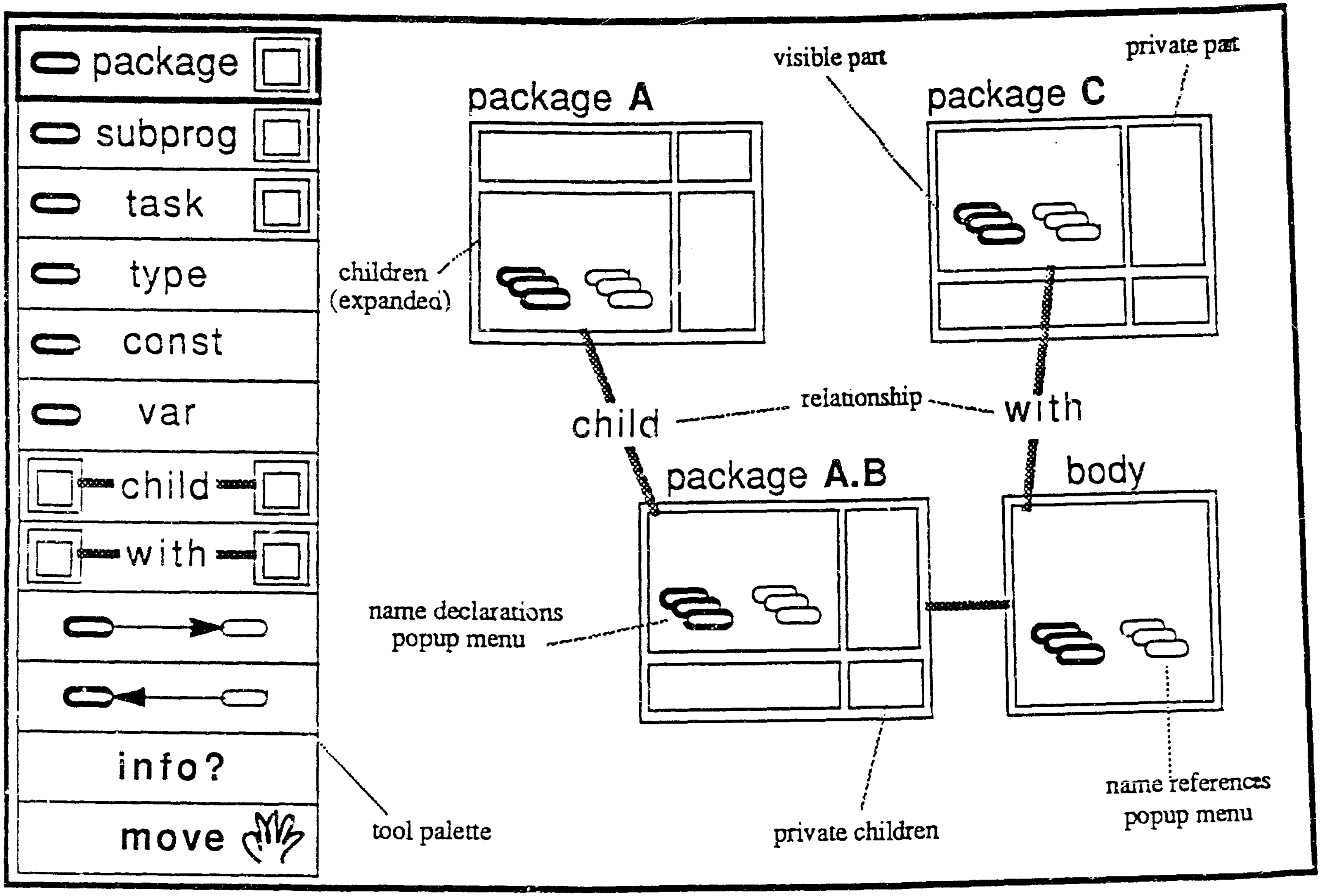


Figure 2: System Architecture View

Participant Briefing
Robert Dewar
New York University

The GNAT project:

A retargetable compiler for Ada9X

Robert Dewar

Edmond Schonberg

New York University

GNAT:

the GNU-NYU-ADA-TRANSLATOR

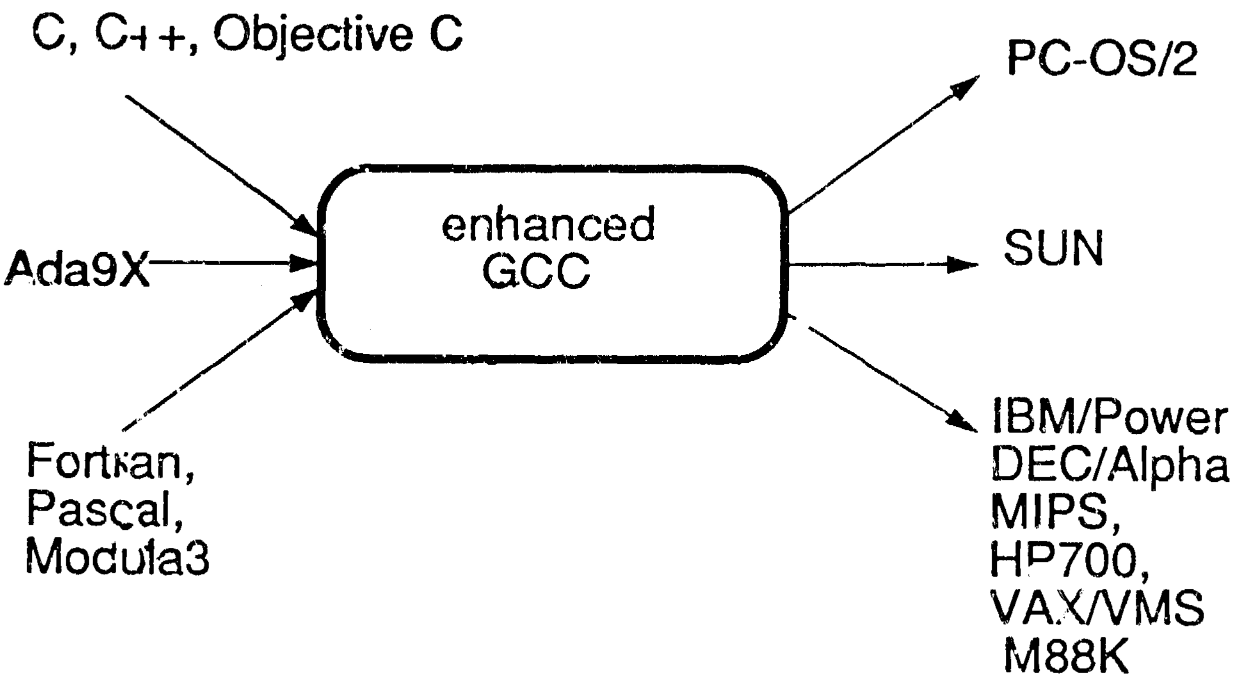
- Goal: a high-quality Ada9X compiler to be made freely available, with sources, to the Ada community.
- To be distributed by the Free Software Foundation as part of the GCC compiler system.
- June 1993: compiler bootstrap.
- September 1993: object oriented features demonstrated and used at TriAda tutorials.
- First release: December 1993 (Ada Lovelace's birthday).

The Uses of GNAT

- An early teaching tool.
- A laboratory for language design.
- A Multilanguage programming environment for Ada 9X, C++ , Fortran.
- A testbed for optimization techniques.
- A compiler for the millions (...who have a Unix box or an OS/2 machine)
- A research tool for programming multiprocessors in Ada9X

GCC

- Product of the Free Software Foundation (Richard Stallman, guiding light).
- A suite of high quality compiler tools, released with sources, under the GNU public license (GPL).



C, C++, Objective C

Ada9X

Fortran,
Pascal,
Modula3

enhanced
GCC

PC-OS/2

SUN

IBM/Power
DEC/Alpha
MIPS,
HP700,
VAX/VMS
M88K

187 targets for GCC (current GNAT targets shown)

GNAT : an open Ada9X Technology

- **Bindings** to open systems interfaces: X, Motif, SQL, POSIX, Mach...
- **Interoperability** and tight coupling with other languages (C, C++, Fortran).
- **Integration** with CASE tools and programming environments, thanks to a radically simplified, source based, Ada library design
- **Extensibility:** source distribution encourages experiments with language extensions.

TWO PRIMARY NEEDS

- INEXPENSIVE INTRODUCTORY COMPILER WITH DECENT ENVIRONMENT RUNNING ON ENTRY LEVEL HARDWARE (CF TURBO PASCAL)
- HIGH END RESEARCH TOOLS, INCLUDING ADA COMPILERS, DEBUGGERS WITH EASILY AVAILABLE, MODIFIABLE SOURCE CODE (CF GCC, G++)

TWO PRIMARY NEEDS (CONT)

- GNAT IS TARGETTED TO THE SECOND GOAL (BUT MAY COME CLOSE TO MEETING FIRST GOAL)

ADDRESSING QUESTION 1

- THE TOP DOWN VIEW
- PEOPLE WILL TEACH WHAT THEY KNOW AND WHAT THEY LIKE AND WORK WITH
- GIVING Ada A LEGITIMATE RESEARCH PRESENCE WILL FILTER DOWN TO ITS BEING TAUGHT AND SUPPORTED

ADDRESSING QUESTION 2

MAJOR NEEDS

- AVAILABILITY OF Ada BASED TOOLS FOR SUPPORTING RESEARCH
- SUPPORT FOR Ada RELATED RESEARCH (NSF POLICY)
- RESPECTABILITY OF Ada IN CS (CF COBOL)

ADDRESSING QUESTION 3

- IT IS NOT ENOUGH TO TEACH Ada AND GENERATE STUDENT ENTHUSIASM
- COMMERCIAL Ada TOOLS AND ENVIRONMENTS MUST BE THERE TO SUPPORT THE NEEDS OF COMMERCIAL USE

EXAMPLES OF POSSIBLE USE OF GNAT

- RESEARCH INTO REAL TIME LANGUAGES (ALL NOW BASED ON G++)
- GANNON'S WORK ON CONCURENCY MAPPING (G++ BASED)
- SUNY BINGHAMTON WORK ON ORDER OF ELABORATION (Ada, BUT NO IMPLEMENTATION

ADA, GNAT and HPCC

- Concurrent programming features of Ada are a promising model for large-scale parallel programming across architectures.

- GNAT will allow this promise to be realized by spreading the use of Ada9X in the research community, and by stimulating experiments:

  - Use of tasking and data synchronization primitives for control of fine-grained parallelism.

  - Use of distribution features to control coarse-grained parallelism (e.g. implementation of Linda)

  - Bindings to the ARPA common runtime support for high-performance parallel languages.

  - Language extensions and optimizations to exploit specialized parallel architectures

Participant Briefing
CDR Lindy Moran, USN
NCIC

(Presentation materials were submitted in handwritten form, a typed copy was created by the Ada Dual-Use Workshop staff.)

Dr. Lindy Moran
Moranl @ ajpo.sci.cmu.edu
(202) 282-0833

- Catching the wave
    - Compilers
    - Environments
    - Libraries of pkgs
    - Textbooks

- Getting in front of the wave
    - Link to exciting "hook"
      (like UR, etc.)
    - Use of multimedia in environments, libraries on CD? verbal help? animation?

- Currents in the wave
    - Academic "irrelevancies"
        - Modularity
        - Maintainability
        - Portability
        - Reliability
    - Changing our CSE paradigm
        - Emphasis on pgm in large
            - reuse, etc
            - concurrency
        - Emphasis on specification part of lifecycle
        - Emphasis on design
    - Hardware regmts
    - Bindings

Participant Briefing
Richard Bolz
Ventura Technology

Ada Dual-Use Workshop

Position Paper, Academia Panel
Richard E. Bolz
Ada Consultant

I served as an Associate Professor of Computer Science for many years at the USAF Academy. Following my retirement from the USAF in 1985 I began a private independent consulting practice primarily involved with Ada training. Counting academic courses, tutorials and professional seminars, I have taught the language over 100 times since 1979. I have given faculty and graduate student Ada seminars at several major universities.

My information regarding Computer Science faculties might be a bit out-of-date. it is gleaned from my experience as a faculty member, my discussions with faculty members of other institutions and anecdotal research formed by discussing university experience with my current students, most of whom are experienced professionals in the computing field.

I was involved in the Ada training and development for the STANFINS project conducted by CSC for the US Army at Ft. Benjamin Harrison, Indianapolis, Indiana. This project redesigned and converted the army's Standard Financial System from COBOL to Ada The project was accomplished by a staff of over 90 software developers most of whom knew only COBOL prior to the project. The system was delivered under budget and virtually on time. The following project, STARFIARS, an inventory and accounting system, has made good use of the lessons learned and technologies developed in the STANFINS project.

Projects fail for sociological reasons, not technological reasons
Bolz's Law

A corollary to this law might be *transition to using Ada in the commercial world is a sociological problem, not a technological problem.* I begin with the premise that there are no technological barriers to the increased use of the Ada language in the commercial world. Most of the issues discussed herein are people issues.

1. What actions can we take to make Ada one of the primary teaching languages within major colleges and universities?

a. There must be economical, efficient translators available. Until we can provide a capability like Turbo Pascal or even WATFOR, we will not be able to accommodate the mass requirements of an introductory course in CS. If such translators exist, we should find a means of getting them to the universities at little or no cost. This is the only suggestion that is primarily technological.

b. There must be strong faculty support. (see # 2)

c. We must be able to show, convincingly that Ada is a legitimate language for teaching. (see #2). We must be well-versed in all of the arguments that might be raised.

1). Gather information from schools that currently use Ada. I assume that ASEET already has much of that information available.

2). Develop a sample curriculum showing the benefits of using Ada from course to course.

3). Establish a team or teams which could tour selected departments offering seminars, discussing proposals with faculty and gathering data regarding perceived stumbling blocks.

2. What initiatives can we mount to increase Ada's appeal, adoption and popularity within the academic community?

a. Many university departments honor research above teaching. The clearest path to adopting Ada as a teaching language would be along the research lines.

1). Issue an RFP for schools to test the viability of using Ada in at least a three course curriculum. This curriculum should probably consist of Introduction to Programming, Data Structures, and Algorithm Development. If the language is as effective a teaching language as we predict, those schools will likely continue with the use of Ada in these and also more advanced courses.

2). Insist that certain research projects given to selected universities be accomplished in Ada.

b. In many CS departments, the curriculum is only nominally integrated (if at all). That is, software engineering courses tend to tout the language-independent nature of life-cycle approaches. They often overlook the benefits of certain languages over others relative to such issues as abstraction and information hiding. Students are rarely introduced to the concept of a compilable specification.

c. It is difficult to experience principles of programming-in-the-huge at the university level. Semester limitations allow only examples of programming-in-the-small, medium or, in the case of multi-semester courses, programming-in-the-large. Provide a suite of case studies which exemplify the scale-up difficulties of very large systems and the benefits of the Ada language in dealing with such difficulties.

d. Most faculty members have not taken the time to learn the language. One approacn to solving this is to have a teach-the-teachers course, totally government funded, for CS faculty members of targeted universities.

e. Many schools are beginning to focus on object-oriented methodologies. It might be wise to wait for Ada 9X before trying to 'infiltrate' these schools.

3. **What investments can we make to increase the commercial use of Ada?**

a. A partitioned approach will probably work best.. We should target the commercial world sector-by-sector. That is, the plan for increased use should probably vary from application area to application area. The optimum approach for the MIS sector might not be the optimum approach for the simulation sector.

b. We should not try to convince the commercial side to merely 'switch' languages. We must be able to discuss the benefits of the language in terms of the entire life-cycle. Perhaps the best approach would be to focus on certain Ada-oriented CASE approaches.

c. We must be able to show successes, from the DOD world, in the particular sector we are targeting. The commercial side is very results-oriented. If we target the MIS sector, then we might show the success of STANFINS and STARFIARS which dealt with US Army finance, accounting and inventory.

d. While the absence of DOD success stories in the sector would be damaging, the presence of such stories, in and of itself, will not be convincing. We must take steps to actually build success experiences in the commercial area itself. One approach would be to fund and partially staff parallel development of selected projects.

e. Realize that we will be dealing with a great amount of resistance-to-change. Professional software developers have gotten comfortable with a certain way of doing business. We must, somehow, minimize the trauma of transition.

1). One approach would be to provide a 'work alike' environment so that the software developers have the assurance that all the tools they are used to using have some kind of counterpart in the new environment. This allows a smooth transition, one new idea/tool at a time. An example would be to provide a CICS-like capability for the MIS sector. CICS is available on HP systems.

2). Another approach is to provide functional equivalency at a higher level. This might occur if we introduce Ada within the context of a CASE environment that includes such capabilities as automatic capturing of requirements via screen 'painters' and also automatic generation of Ada software and documentation from the captured requirements.

f. If the target sector is one with enhanced support from Ada 9X (MIS for example) then the appeal for transition will be made easier if we wait for Ada 9X to become a reality. Of course, we would not have the success stories to share for a while.

g. There is a CASE environment in existence that builds on the success of the STANFINS and STARFIARS projects. A small company, Ventura Technology International, has developed its Advanced System and Software Engineering Repository and Toolset (ASSERT). This environment, while application and language independent, has been shown to be effective when used on MIS systems written in Ada. It includes capabilities for interactive capture of design information via screen painters, automatic generation of Ada code, and automatic generation of supporting documentation. It is hosted on HP-UX, HP's Open System Environment(OSE).

h. There is an increasing interest in object-oriented methods. We should have a working forum within certain selected sectors (MIS for example) with the intent of designing and even implementing class libraries to support various activities within the sector.

i. Regarding the COBOL community: It has been my experience that COBOL programmers have no particular difficulty making the transition to Ada. There will be, of course, some initial resistance-to-change. That is easily overcome by validating the students' experience and treating them as the professionals they are.

4. What steps can we take to increase the pool of entry level personnel with Ada skills, knowledge, and experience?

a. This will partially take care of itself if we have successfully accomplished the goals of numbers1 thru 3. Many of those in government service with Ada experience will migrate to the commercial sector as the open marketplace has its effect. This, of course, will force us to reconsider the question in light of increasing the pool for DOD.

b. Vigorously target smaller schools, including community colleges. Often these schools have very few CS courses available. They might appreciate a 3-course curriculum (see 2.a) with slides, case studies, examples, lesson plans etc.

c. Provide a forum for these smaller schools and local industry to enter into symbiotic relationship. The local industries can provide adjunct faculty, advice on curriculum and entry-level employment for graduates from the institutions.

Conclusion:

Primary suggestion for success in the MIS commercial sector:

Migrate STANFINS to an OSE and deploy

1. What actions can we take to make Ada one of the primary teaching languages in major colleges and universities?

- Assure availability of efficient, affordable translators
- Gather current information
- Offer sample curricula
- Establish touring teams

2. What initiatives can we mount to increase Ada's appeal, adoption and popularity within the academic community?

- Take the research path
  - RFP for viability testing
  - Mandate Ada for government purchased COTS
- Provide a forum for curriculum integration
- Provide means of appreciating programming-in-the-huge
- Provide government funded education for faculty members

3. What investments can we make to increase the commercial use of Ada?

- Target certain sectors (MIS, Simulation etc.)
- Take a life-cycle approach
- Show success stories
  - DOD experience
  - Fund and staff parallel commercial efforts

3. What investments can we make to increase the commercial use of Ada?

- Encourage development of COTS products with functionality equivalent to that currently used within the sector (CICS for MIS)
- Encourage use of existing CASE environments
- Emphasize Ada 9X annexes (MIS)
- Provide forum for developing class libraries

4. What steps can we take to increase the pool of entry level personnel with Ada skills, knowledge and experience?

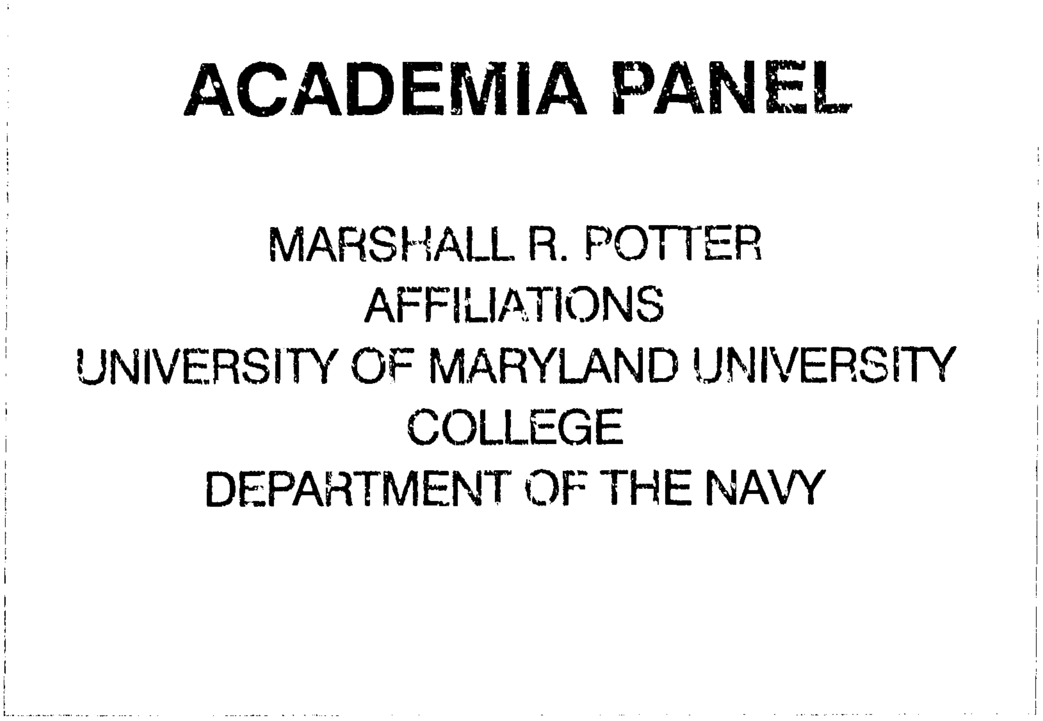
- Steps 1 thru 3 will partially answer the question
- Vigorously target smaller schools
- Encourage symbiotic relationship between small schools and local industry

STANFINS-R

- Redesign of US Army's STANdard FINancial System
- Over 2 million Ada SLOC
- Over 80 COBOL programmers developing in Ada
- Average staff size of 150
- 6 times industry norm regarding productivity (SLOC/staff month)
- 75% automatically generated from repository information
- Documentation automatically generated from repository information
- Toolset written in Ada to support entire life-cycle
- ASSERT/VTI

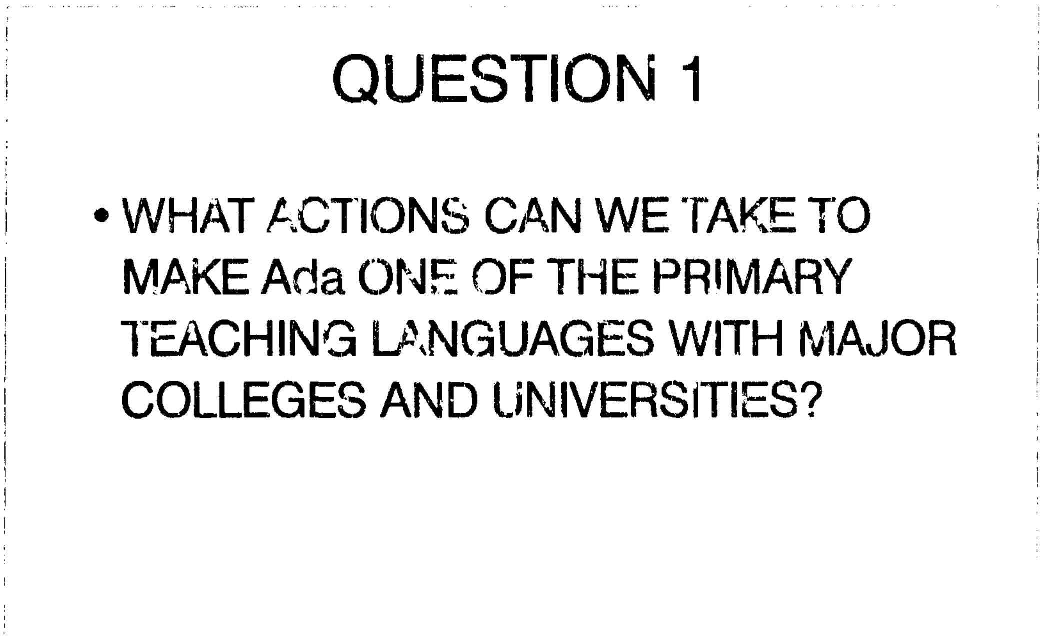
Participant Briefing
Marshall Potter
NISMC

(Presentation materials were submitted in handwritten form, a typed copy was created by the Ada Dual-Use Workshop staff.)



ACADEMIA PANEL

MARSHALL R. POTTER
AFFILIATIONS
UNIVERSITY OF MARYLAND UNIVERSITY COLLEGE
DEPARTMENT OF THE NAVY



QUESTION 1

- WHAT ACTIONS CAN WE TAKE TO MAKE Ada ONE OF THE PRIMARY TEACHING LANGUAGES WITH MAJOR COLLEGES AND UNIVERSITIES?

SUGGESTED ACTIONS

- DEVELOP SOME QUALITY TEXTBOOKS
- PROVIDE SOME TEACHING TOOLS & AIDS
  - LEGACY CODE
  - CAI
  - VISUALIZATION TOOLS
  - VIDEOS
  - CASE TOOLS TO SUPPORT Ada

SUGGESTED ACTIONS (CONT)

- PROVIDE COMPILERS & ENVIRONMENTS AT NO/VERY LOW COST

QUESTION II

- WHAT INITIATIVES CAN WE MOUNT TO INCREASE Ada's POPULARITY WITHIN THE ACADEMIC COMMUNITY?

POSSIBLE INITIATIVES

- DEVELOP SOME TRANSITION WORKSHOPS
  - PROVIDE TOOLS/MATERIALS
  - PROVIDE INSTRUCTOR TRAINING
- PROVIDE FOR FREE/VERY LOW COST COMPILERS AND ENVIRONMENTS
- PROVIDE FOR A NATION-WIDE PROGRAM TO IMPROVE THE COMPUTER SCIENCE EDUCATION PROCESS (INFRASTRUCTURE)

POSSIBLE INITIATIVES (CONT)

- GET INDUSTRY INVOLVED TO DEMAND Ada EDUCATION FOR THEIR EMPLOYEESOPEN UP REUSE LIBRARIES FOR EDUCATIONAL USE
- OPEN UP REUSE LIBRARIES FOR EDUCATIONAL USE
- GET UNIVERSITIES INVOLVED IN ON-GOING DOD/GOVERNMENT Ada PROGRAMS

QUESTION III

- WHAT INVESTMENTS CAN WE MAKE TO INCREASE THE COMMERCIAL USE OF Ada?

POSSIBLE INVESTMENT

STRATEGIES

- DETERMINE WHY Ada IS NOT THE LANGUAGE OF CHOICE FOR SPECIFIED DOMAINS?
- DETERMINE STRENGTHS & WEAKNESSES OF Ada IN THOSE DOMAINS.

POSSIBLE INVESTMENT

STRATEGIES (CONT)

- DETERMINE IF INVESTMENT HAS A POTENTIAL PAYOFF, HOW MUCH IS THE PAYOFF, AND HOW LONG WILL IT TAKE TO ACCOMPLISH?

QUESTION IV

- WHAT STEPS CAN WE TAKE TO INCREASE THE POOL OF ENTRY LEVEL PERSONNEL WITH Ada SKILLS, KNOWLEDGE AND ABILITIES?

STEPS TO TAKE

- NEED TO GET DOD AND OTHER GOVERNMENT R&D COMMUNITIES BEHIND THE Ada MANDATE AND START REQUIRING Ada TO BE USED.
- INITIATE TRAINING AT HIGH SCHOOLS
- INFLUENCE YOUTH GROUP PROGRAMS SUCH AS THE BOY SCOUTS/GIRL SCOUTS

STEPS TO TAKE (CONT)

- WRITE HOBBY LEVEL BOOKS USING Ada
- NOT ENOUGH TO HAVE JUST Ada COURSES

Participant Briefing
Eugen Vasilescu
Grumman Data Systems

Information Systems Curriculum Observations

Eugen N. Vasilescu

DISA Ada Dual-Use Workshop - October 19-20, 1993

Computer Education Tracks

- Computer Science (CS) Education
  - College of Liberal Arts/Engineering Schools
  - ACM based Curriculum
- Information Systems (IS) Education
  - Schools of Business
  - AACSB based Curriculum; general requirements in
    - Accounting
    - Finance
    - Business Law
    - Marketing

Core AACSB IS Curricculum

- Two programming Courses
  - Normally COBOL I and II
- Systems Analysis Course
  - Data Flow Diagrams
  - Various SD Methodologies
  - Business case oriented
  - CASE component
- Database Course
  - Relational model emphasized
  - Semantic and OO models used for
    - Conceptual schema design
    - Business rules/modeling
  - Integration with previous core courses

Notes on COBOL

- COBOL mostly associated with mainframe environment
- Perceived as legacy programming language
- No religious attachment to it
- Weak support for software engineering
- Well integrated within mainframe environment
- Good file handling capability
- Good financial computation features

Ada 83 Potential For IS curriculum

- Good support for Software Engineering
- Excellent portability across platforms
- Good potential for business case development
- Good life-cycle characteristics in large systems
- Weak interfaces and file handling capabilities
- Weak support for financial computations

Ada 9X Potential in IS Curriculum

- Excellent support for Software Engineering
- Excellent portability across platforms
- Enhanced potential for business case development
- Assumed improved life-cycle characteristics in large systems
- Excellent interfaces to external environments
- Good support for financial computations

Recommendations

- Develop an AACSB compliant curriculum for Ada 9X
- Develop AACSB targetted course materials
- Develop integrated Ada 9X centered business cases for easy articulation with AACSB follow-on courses
- Develop Ada 9X bindings to open systems standards

Participant Briefing
Timothy Long
Ohio State University

Promoting the Use of Ada in Computer Science Curricula

Timothy J. Long

Ohio State University

To Increase Use of Ada in Academia:

- Supplant Pascal
- Compete with the move to C/C++

Necessary to Achieving Both Are:

- Wide availability of good materials
    - Textbooks
    - Programming assignments
    - Software libraries
- Wide availability of "student" compilers
    - Inexpensive
    - Fast
    - Very friendly
    - Run on variety of "low-end" equipment

A Philosophical Stand on the Computer Science Undergraduate Curriculum:

- The backbone of the curriculum is a REDESIGNED sequence of courses, CS1, CS2, CS3, . . . , on programming and software development/engineering.
- THE ENTIRE SEQUENCE IS FIRMLY GROUNDED IN SOFTWARE ENGINEERING PRINCIPLES.
- The entire sequence is uniform, presenting a consistent view of what software is and how it should be developed.

Then, to Supplant Pascal:

- From DAY 1 OF CS1 emphasize software engineering principles that require features not present in Pascal:
  - Object-oriented design
  - Reusable and extendible software components
  - Information hiding beyond procedural abstraction

Proof of Principle:

- Long-term studies can track students measuring:
    - Student attitudes towards the principles and practices taught
    - Whether students apply these principles and practices in other courses
- Positive responses in such studies would imply that students have the EXPECTATION to use similar languages and practices in the COMMERCIAL work place.

Please Note - This Proposal is EXPENSIVE:

- It requires consistent, uniform materials over a sequence of courses.
- It is unlikely that current CS1 materials are appropriate, so the effort starts with CS1.
- Multiple faculty will need to be involved.
- Tracking studies must be well designed and scientifically rigorous in order to carry weight.
- The entire project has to be funded for the long term.

But, a BIG study showing good results could very well have a BIG PAYOFF.

Meeting the C/C++ Challenge:

- Promoting Ada as the second language in the curriculum probably will not work.
  - There is “downward” pressure for C++.
  - Students are susceptible to “C++ fever”.
- C should not pose any serious competition.
  - There is very little support in C for good software engineering principles.

C++ Will Be Strong Competition in Academia:

- Object-oriented nature of C++ makes it attractive.
- There is "downward" pressure in the curriculum for C++.
- There is pressure from the textbook industry for C++ materials.

Consider a Co-existence Strategy:

- In a longer course sequence, there is plenty of room for both languages. Ada would be pedagogically preferable for the first language.
- Consider comparative studies measuring the relative strengths of each language and effects of using both in the curriculum.

Additional Point: How Will Ada 9X Impact the C++ Issue?

To Summarize:

- For use in academia, the strengths of Ada are its support for
  - object-oriented design, and
  - good software engineering practices.

- Fund projects that are based on this view of the strengths of Ada, and that are based on
  - making object-oriented design and
  - software engineering principles and practices

  the cornerstone of the undergraduate sequence in programming and software engineering, beginning with CS1.

- Fund long-term projects having the greatest potential for
  - demonstrating the effectiveness of this approach to the undergraduate curriculum, and
  - developing the corresponding materials for such a curriculum.

- Seize the initiative now, or lose it to C++!!

Participant Briefing
Philip Miller
Carnegie Mellon University

Philip Miller

Director of Introductory Programming

School of Computer Science

Carnegie Mellon University

plm@cs.cmu.edu

What actions can we take to make Ada one of the primary teaching languages within major colleges and universities?

- Important Dimensions of Languages

  Ubiquity

  Easy to Learn

  Theoretical Foundation

  Rapid Prototyping

  Essentially Cool

  High Reliability

- Dimensions of Language x Environment
  - Easy to Teach Principles
    - Round Up the Usual Suspects
    - Emphasize Ada's Unique Perspective
  - Gentle Learning Curve for Novices
  - Smooth Transition to Professional Tools

What initiatives can we mount to increase Ada's appeal, adoption and popularity within the academic community?

Support of Quality Activities

Continued Sensitivity to Change

Stable Employment Picture

Realistic Expectations

Course Materials Telling the Ada Story

Participant Briefing

Jon Spear
Naval Postgraduate School

(Presentation materials were submitted in handwritten form, a typed copy was created by the Ada Dual-Use Workshop staff.)

JON SPEAR
(Spear @ CS.NPS.Navy.MIC)
Naval Postgraduate School

EXPAND DEVELOPMENT, DISTRIBUTION AND USE OF Ada-BASED

- TOOLS AND ENVIRONMENTS
  - CAPS (COMPUTER-AIDED PROTOTYPING SYSTEM)
  - AdaSAGE
    - GNAT
    - ADA GENIE
    - ADA-ED
    - BINDINGS
- COURSEWARE
  - TEXTS THAT EFFECTIVELY APPLY Ada
  - CURRICULUM DEVELOPMENT

TEACHING HARD REAL-TIME ADA SOFTWARE DEVELOPMENT AT NPS

Courses employing Ada:

- **CS2970 “Structured Programming with Ada”**
- **CS3300 “Data Structures”**
- **CS3460 “Software Methodology”**
- **CS3050 “Software Development for Combat Systems”**
- **CS4150 “Programming Tools and Environments”**
- **CS4500 “Software Engineering”**
- **CS4520 “Advanced Software Engineering”**
- **CS4530 “Software Engineering with Ada”**

Other supporting Courses:

- **CS4540 “Software Testing”**

TO ENCOURAGE ACADEMIC USE OF ADA

* New Textbook that supports Ada Software Engineering

- Software Engineering with Abstractions
  Berzins & Luqi, Addison-Wesley 1991
- Systematic Software Development with integrated Requirements Specification
- Illustrated with large examples in Ada covering the entire life cycle
- Approach has been successfully used at
  Naval Postgraduate School (7 years)
  University of Minnesota (5 years)
  and dozens of other universities

* State of the art tools to support syntax-directed editing, automatic type checking, and Ada interface code and automatic test oracle code generation from the Requirements Specs.

* Current tools and materials available via anonymous ftp from taurus.cs.nps.navy.mil at /pub/SPECTOOLS

* Recommendations:

- provide funding support to refine and distribute the tools

CAPS

FOR AUTOMATED REAL-TIME SOFTWARE DEVELOPMENT

OBJECTIVES

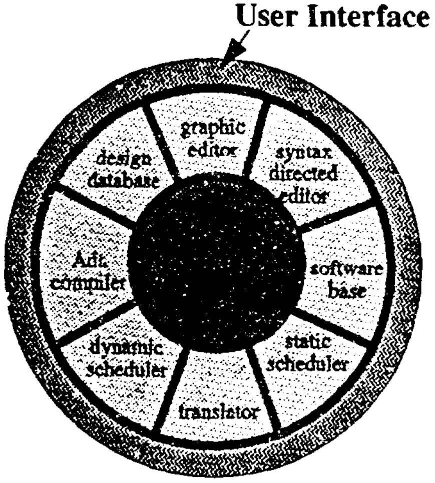
Assist DoD program managers to rapidly evaluate requirements for military real-time control software using executable prototypes and to test and integrate completed subsystems through evolutionary prototyping.

ISSUES

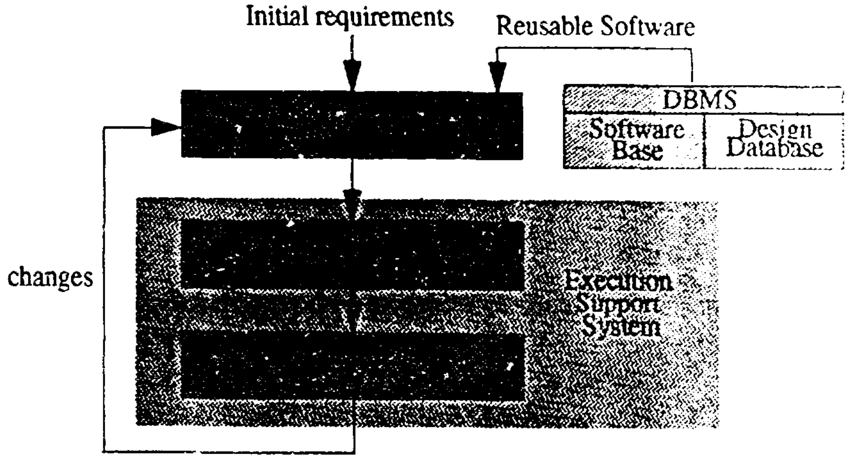
- Inability to accurately and completely specify requirements for real-time software systems results in poor productivity, schedule overruns, unmaintainable and unreliable software.
- CAPS provides a capability to quickly develop functional prototypes to verify feasibility of system requirements early in the software development process.
- CAPS supports an evolutionary development process that spans the complete life-cycle of real-time software.

APPLICATIONS / PAYOFF

- Formulate/validate requirements via prototype demonstration and user feedback
- Assess feasibility of real-time system designs
- Enable early testing and integration of completed subsystems
- Support evolutionary system development, integration & testing
- Reduce maintenance costs through systematic code generation
- Produce high quality, reliable and flexible software
- Avoid schedule overruns



CAPS Advanced Rapid Prototyping Environment



Iterative Prototyping Process in CAPS

AdaSAGE

APPLICATION DEVELOPMENT TOOLS & LIBRARY

- INTERACTIVE SCHEMA & SCREEN EDITOR PRODUCES DATA DICTIONARY USED BY OTHER TOOLS
- EXTENSIVE SUPPORT LIBRARY
  - RDBMS
  - GKS
  - SQL
  - PRESENTATION GRAPHICS
  - UTILITIES

WIDE USE BY DOD (80+ SITES), INDUSTRY (50+ SITES)

- APPLICATIONS
  - MIS
  - SCIENTIFIC & ENGINEERING
  - MANY OTHERS

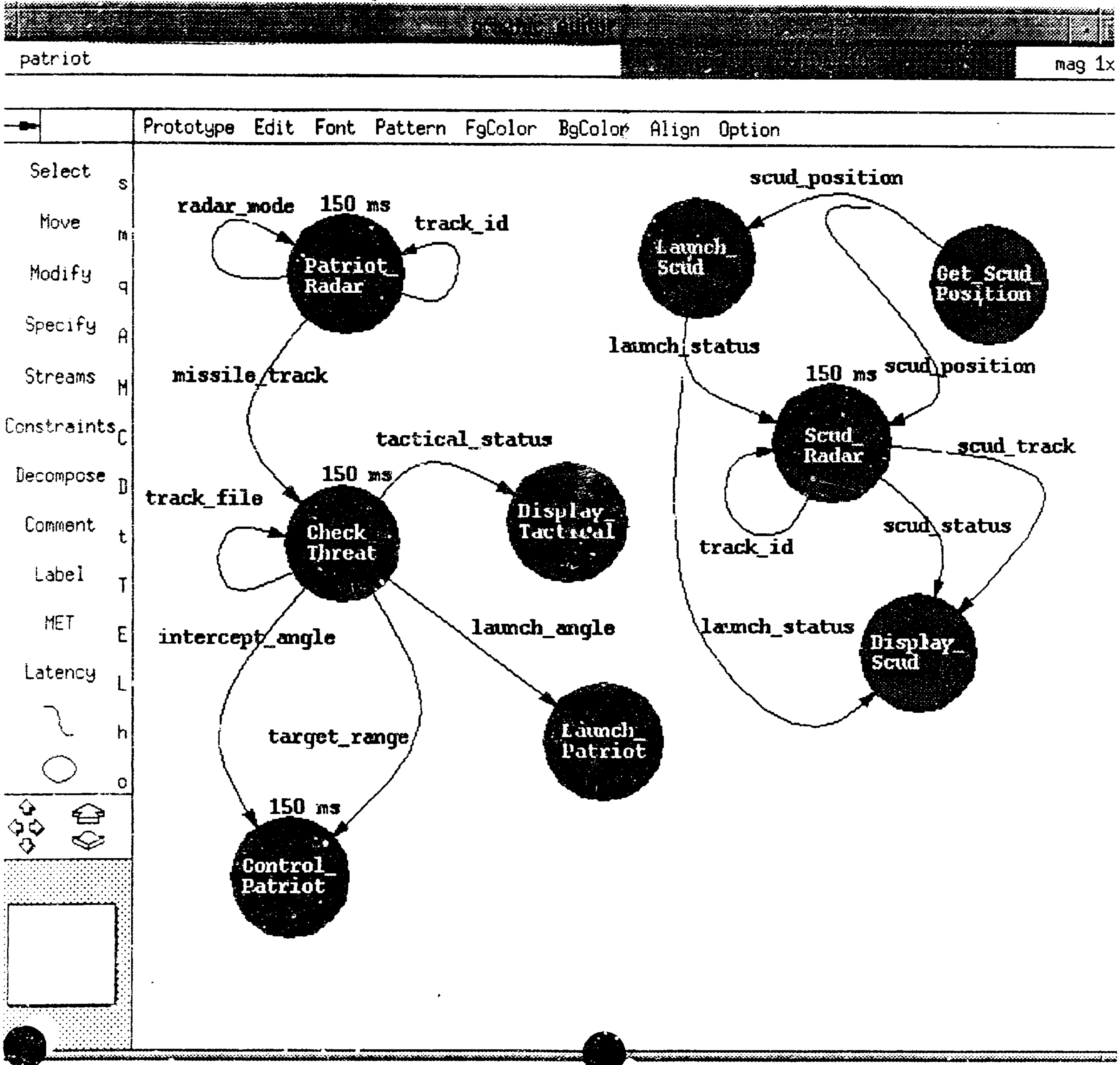
ACADEMIC USE AT OVER 180 SCHOOLS

- PROTOTYPING AND APPLICATION DEVELOPMENT
- DEMONSTRATES EFFECTIVE RE-USE
- PERMITS MORE REALISTIC PROJECTS

RECOMMENDATIONS

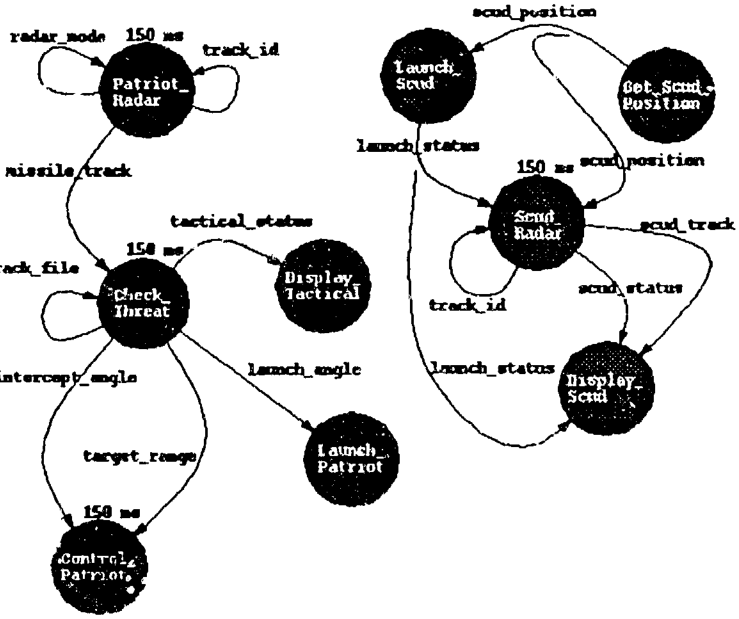
- IMPROVE AVAILABILITY TO SCHOOLS
- UPDATE FOR Ada 9X
- IMPROVE PORTABILITY

PSDL Implementation Graph:



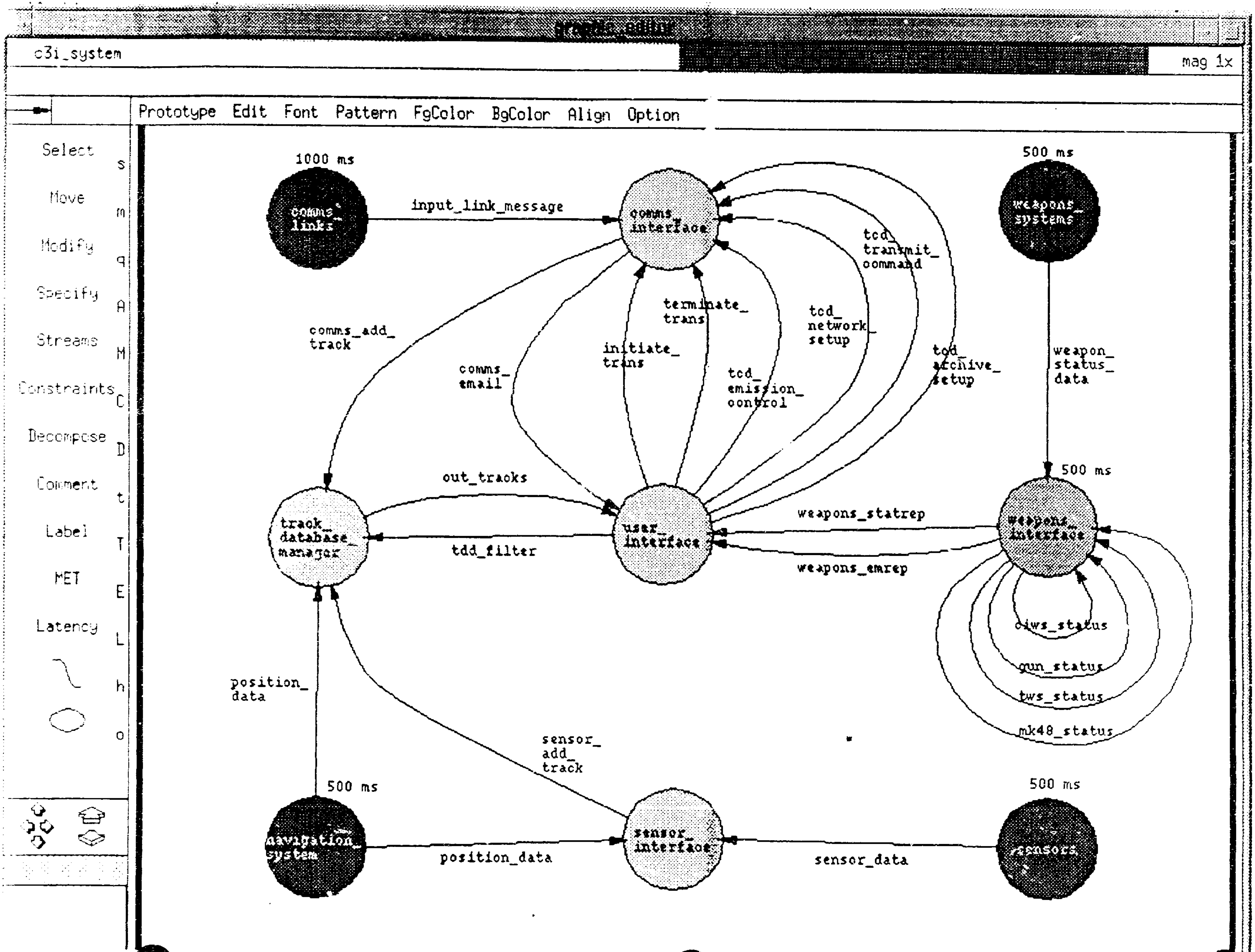
DETAILED EXAMPLE

```
OPERATOR patriot
SPECIFICATION
    DESCRIPTION
    {Patriot Missile Defense System simulation.
     Simulates Patriot intercepting a Scud missile.}
END
IMPLEMENTATION
    GRAPH
```

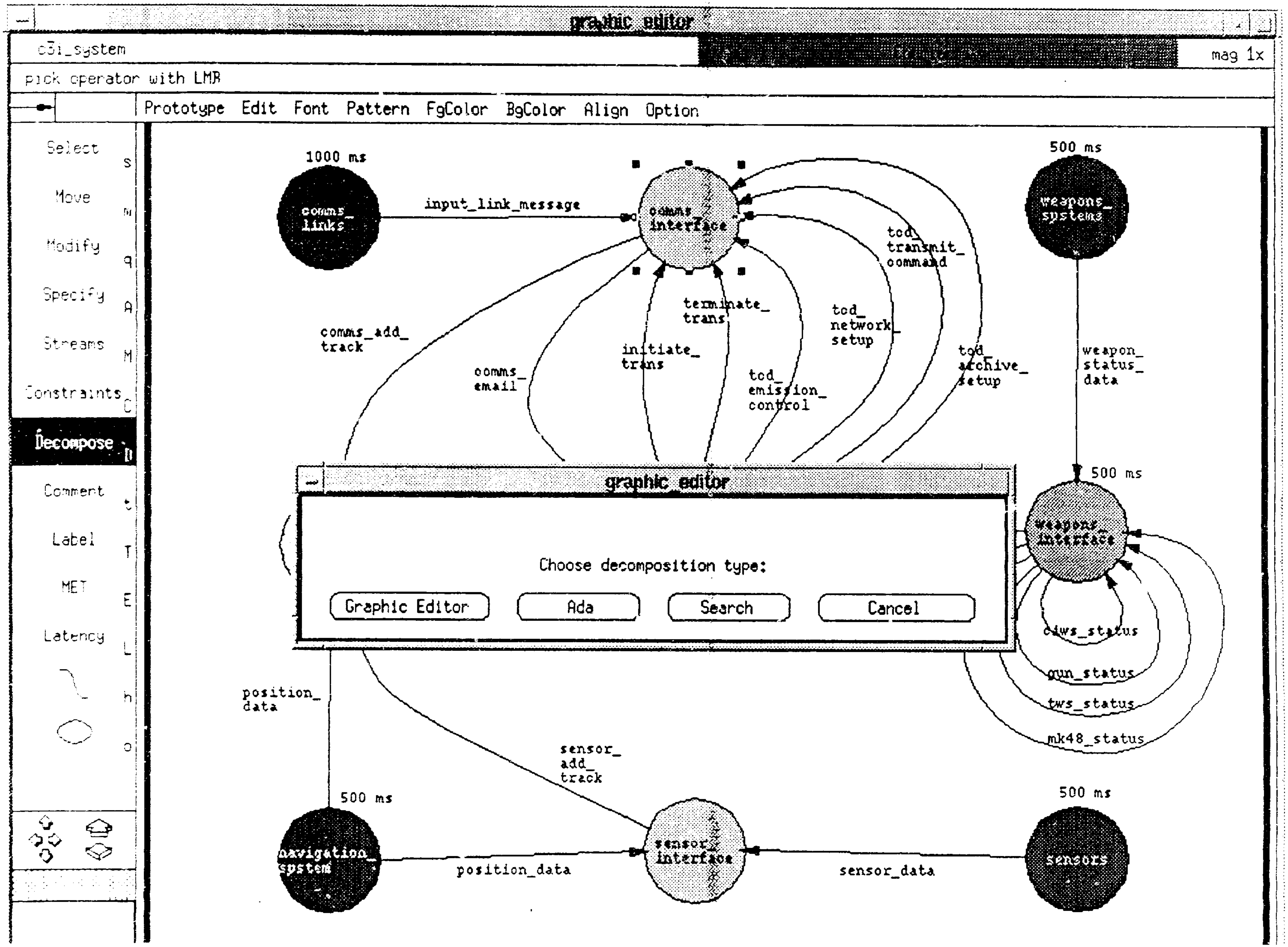


```
    DATA STREAM
    -- Type declarations for the data streams in the graph go here.
    CONTROL CONSTRAINTS
    OPERATOR Patriot_Radar
    PERIOD 800 MS

    OPERATOR Scud_Radar
    PERIOD 800 MS

    OPERATOR Display_Tactical
    TRIGGERED BY SOME
     tactical_status

    OPERATOR Control_Patriot
    TRIGGERED BY ALL
     intercept_angle,
     target_range

    OPERATOR Launch_Patriot
     TRIGGERED BY ALL
      launch_angle

    OPERATOR Check_Threat
     PERIOD 800 MS
     OUTPUT
      intercept_angle
      IF NOT (intercept_angle = 0.0)
     OUTPUT
      target_range
      IF NOT (intercept_angle = 0.0)
     OUTPUT
      launch_angle
      IF NOT (launch_angle = 0.0)
END
```

DETAILED EXAMPLE

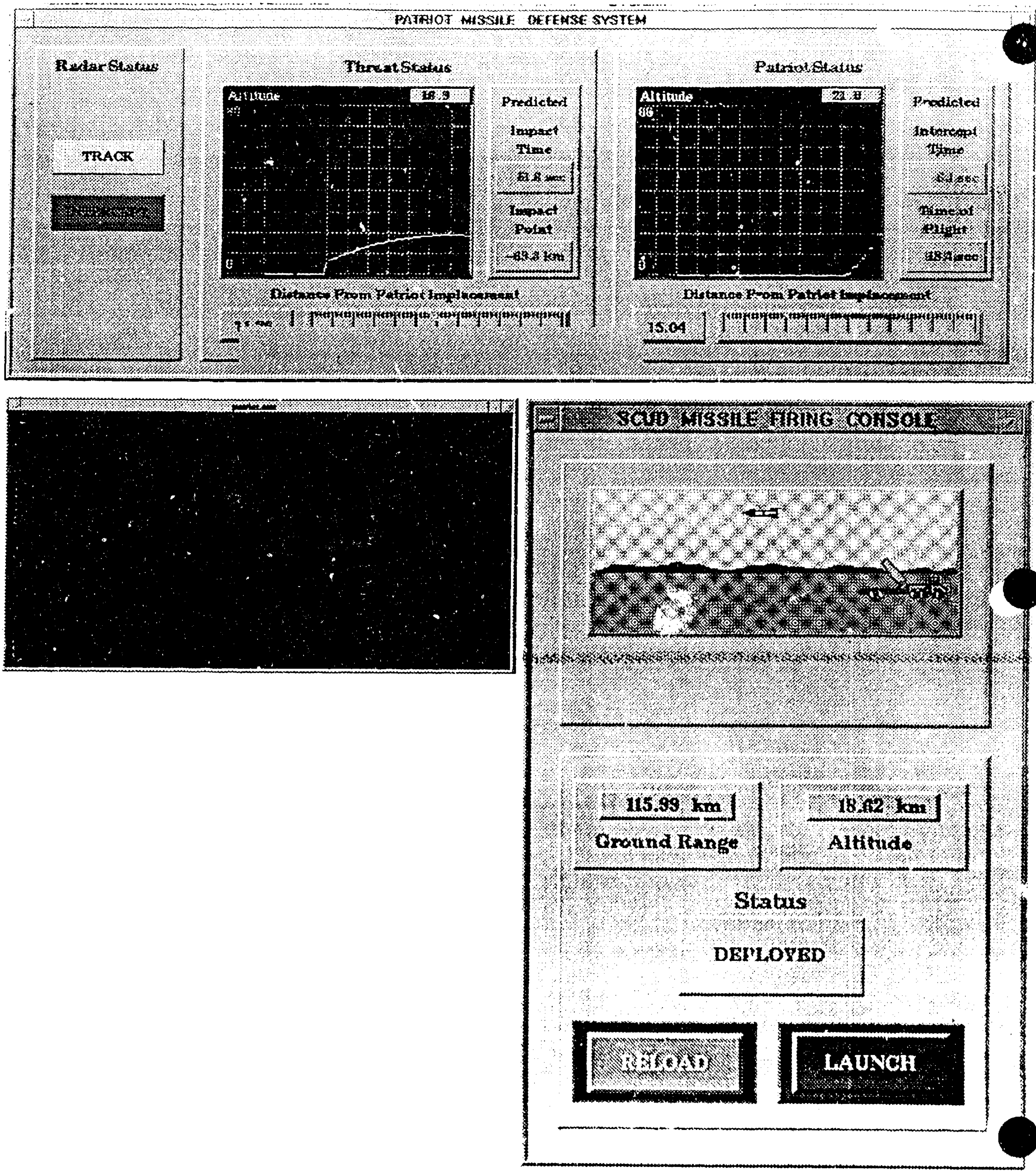


PSDL DECOMPOSITION:



Choose decomposition type:

DETAILED EXAMPLE



PATRIOT MISSILE DEFENSE SYSTEM

Radar Status

TRACK

Threat Status

Altitude 18.9

Predicted Impact Time 51.8 sec

Impact Point -63.3 km

Distance From Patriot Implacement

Patriot Status

Altitude 23.8

Predicted Intercept Time 5.4 sec

Time of Flight 18.4 sec

Distance From Patriot Implacement 15.04

SCUD MISSILE FIRING CONSOLE

115.99 km Ground Range

18.62 km Altitude

Status DEPLOYED

Participant Briefing
Don Needham
Carnegie Mellon University

Ada in Academia

Don Needham
Computer Science Instructor
U. S. Naval Academy

Used Ada in: Software Engineering,
Operating Systems,
Programming Languages,
Introductory Ada course

1. What actions can we take to make Ada one of the primary teaching languages within major colleges and universities?

A. Give us "Turbo Ada". Invest funds to develop a PC based compiler as user friendly and capable as Borland C.

- You are competing with C++'s support environment, need to be competitive with C++ in terms of tools, availability and price.

- Compiler should have a debugger similar to Borland's with the ability to step through source code line by line and inspect values of variables.

- The compiler should have speed similar to Borland's
  - implement various subsets of Ada if necessary
  - the into course should be as "Borland-like" as possible.

- Reasonable hardware requirements, so can be used in a University with minimal upgrade requirements (if any).

B. Make this "Turbo Ada" compiler free to universities.

- Universities are more prone to try something new if faculty are motivated and "price is right".
- Rational's Apex environment.
  - need to upgrade lab hardware.
  - step in the right direction for upper-level courses.

2. What initiatives can we mount to increase Ada'a appeal, adoption and popularity within the academic community?

A. Do not make Ada's use required by law.

- Forcing people to do something just makes them want to oppose it on principle.

- The reason to use Ada should be because it is the best language choice.

B. Fix the Software Repositories.

- Repositories need to house only tested, reliable Ada code.

- Locating the components needed within a repository should be convenient, simple, fruitful.

C. Develop Ada introduction teams.

- Provide interested universities with a means for acquiring the necessary Ada programming exposure.

- Need to combat the tendency of people sticking to what they already know, opposing change.

3 What investments should we make to increase the commercial use of Ada?

A. Again, do not force Ada's use. Ada needs to (and can) stand on its own merits.

4. What steps can we take to increase to pool of entry level personnel with Ada skills, knowledge and experience.

A. Industry demand drives this equation. Teaching schools need to provide their graduates with skills that are in demand, where the jobs are.

Participant Briefing
Robert Willis
Hampton University

(Presentation materials were submitted in handwritten form, a typed copy was created by the Ada Dual-Use Workshop staff.)

Robert A. Willis, Jr
Hampton University

Questions: My Responses

Pessimistic Response

There is nothing that can be done to increase Ada's use significantly in academia and industry.

Anything that would be attempted could be classified as "too little too late".

There have been many mistakes made in the promugation of Ada. It is impossible to overcome these mistakes. The Edsel was a superior product. Superiority in American culture has never been a key to success.

I have listed to many of the presentations and with notable exceptions (Phillip Miller and Timothy Long) nothing was said that hasn't been said (over and over again) for the past 10 years. What was said is correct but irrelevant.

The Ada community has missed the window of opportunity which was open approximately 3 or 4 years ago. Most C.S. professors were at least receptive to the concept of using Ada in their respective curricular. They are not nearly so receptive now.

C/C++ is bigger than Ada. This language has monentum. Remember the C vs. Modula 2 arguments during the CPM original MSDOS days? Where is Modula 2 now?

Pragmatic Response

The pessimistic response is one I'm sure no one wants to hear. But its outcome is highly probable. The widespread use of Ada will never happen.

But negative response are counter productive, in an attempt to be productive and/or helpful I have prepared more pragmatic responses. short term

1 Ada needs good public relations. Everyone in the Ada community must show (prove) that Ada is the superior product.

2 DOD must do a better P.R. job (much, much, much better). You can't make people accept Ada you must convince them.

3 No DOD (in fact no Government) contract or research grant should be awarded that does not have an Ada component. Academics require funds to do research.

Participant Briefing
Joan Langdon
Bowie State University

(Presentation materials were submitted in handwritten form, a typed copy was created by the Ada Dual-Use Workshop staff.)

1 a) Commercial/Government demand
   Ada Trainer

b) Funds for:
   a) student support
   b) faculty support

   Endow an Ada Fellow for 2 years

2 a) Valid compilers -- low cost
b) Valid compilers -- PC & Mainframe

3 - Ada 9X environments with good software development tools
- Compiler development
- Evaluate compilers with eye to improvement. Do not just validate.

4 Summer Workshops
   Summer Programs like the SICA

The SICA program sponsors undergraduate and graduate summer internships at Goddard Space Flight Center for the purpose of introducing space applications to the students. They typically use Pascal, C, and Fortran.

Students could intern at sites which use Ada for various applications.

- Publish some results showing Ada's safety and reliability characteristics.

- Publish all application areas

- We have received calls about Ada courses
   Near GSFC

- Make grant money available for student support and possibly faculty.

- Commerce Business Daily (What is it?)

Currently at Bowie

- Ada compiler but not Ada 9X yet. The free compiler is being asked for.

- Several faculty members can teach this course.

- Courses which would benefit

  We could easily change from Pascal to Ada at this time.
  C is being pushed.

  Faculty workshops for Ada

  Faculty offered workshops to the community at large

- We are decentralizing somewhat

- We are going towards a combination of the two

Make Turbo Ada like Turbo Pascal available

Work with 2 year colleges to implement Ada in conjunction with 4 year schools.

APPENDIX D

ADA DUAL-USE WORKSHOP
OCTOBER 19-20, 1993

GOVERNMENT PANEL PROCEEDINGS

November 8, 1993

Defense Information Systems Agency
Joint Interoperability and Engineering Organization
Center for Information Management
Software Systems Engineering Directorate

TABLE OF CONTENTS

1.0 INTRODUCTION

This appendix contains the findings, conclusions, and recommendations for the three questions that were posed to the government panel. This data supports the summary of the government panel as described in the Ada Dual-Use Workshop Summary.

2.0 QUESTION #1: **What actions can we take to make Ada the primary tool for developing software systems within your government organization?**

FINDINGS

The following findings were the result of participant briefings:

- Knowledge of Ada's benefits as a language and a software engineering tool is lacking. Program Managers believe there is a higher risk using Ada than other languages.
- Many DoD Program Managers ignore the Ada mandate.
- DoD has failed to ensure the development of appropriate high quality Ada bindings; and they, in fact, do not exist.
- The current Ada policy is unclear. It appears that if a Program Manager believes Ada is not cost-effective, then a waiver is not required.

CONCLUSIONS

Based on the session findings, the following conclusions were made by the panelists:

- Information about Ada successes (defense and commercial) is not reaching government executives making programming language decisions.
- The Ada mandate is not well communicated and adhered to throughout the DoD. This provides a negative example to the software community as a whole.
- DoD commitment to Ada appears weak, or at best, academic.
- Robust, high quality bindings are needed for widely used standards, COTS products, and defacto standards.
- We need to evolve to a single binding standard with multiple implementation capability.

RECOMMENDATIONS

The following recommendations were derived from the findings and conclusions listed above:

- DoD should develop and collect standard metrics that show the relative cost, schedule, and quality advantages experienced in Ada projects.
- OASD (C3I) should recognize Program Managers of Ada intensive projects with a special one or two day annual conference that has as its keynote speaker the ASD (C3I).
- DoD should collect Ada success stories in defense applications and publish articles in publications that reach MIS management and technical personnel (e.g., Business Week, Information Week, Datamation, PC Magazine, PC Week, Byte Magazine). (This recommendation is an appropriate response to question #2 as well.)
- DoD should develop a video that describes the Ada policy, identifies programs using Ada, shows the metrics, teaches the lessons learned, and describes available Ada help. The video should be disseminated widely, including industry, and be required viewing for all software managers in DoD. (This recommendation is an appropriate response to question #2 as well.)
- A focused binding effort should be fostered which includes the following steps:

  - Identify candidates
  - Narrow choice among competitors
  - Stimulate comparative development effort
  - Publish specifications
  - Subsidize prototype implementation
  - Standardize appropriate binding specifications

3.0 QUESTION #2: What investments can we make to increase the commercial use of Ada?

FINDINGS

The following findings were the result of participant briefings:

- Most non-defense software developers (including universities) perceive Ada as a programming language solely for defense applications.
- Among the general programming community, there is not a high level of enthusiasm for Ada training and education.
- DoD has not been proactive in urging vendors to develop and provide Ada bindings.
- Most colleges and universities can't afford the high cost of Ada tools.

CONCLUSIONS

Based on the session findings, the following conclusions were made by the panelists

- Information about Ada successes (defense and commercial) is not reaching industry.
- Information about Ada is not reaching colleges and universities.
- DoD needs to provide encouragement to the academic community and small business companies to implement teaching and training in Ada.
- DoD should motivate commercial product vendors to provide timely, fully functional Ada interfaces.
- DoD should ease the process for waivers/exceptions for interface code to non-Ada applications.

RECOMMENDATIONS

The following recommendations were derived from the findings and conclusions listed above:

- DoD Ada-related events need stronger press coverage.

- A qualifying characteristic for the new AJPO Director should be that he/she have broad appeal and be able to relate to the entire software community, not just DoD weapon systems.

- OASD (C3I) should give a spot award ($500) to DoD authors of articles about Ada in non-Ada specific press.

- DoD should inform colleges and universities of the availability of Ada tools through the Free Software Foundation.

- DoD should continue to support and fund the Free Software Foundation, especially in the area of the Ada compiler with Ada 9X enhancements.

- DoD should donate older versions of Ada tools to colleges and universities.

- DoD should promote Ada within the academic community (universities, junior colleges, high schools) as an excellent tool for teaching software engineering techniques.

- DoD should promote/provide accessibility of compilers to educational institutions.

- DoD should encourage private industry to offer a variety of Ada training courses.

- DoD should emphasize Ada as the primary software language at all military academies.

- DoD should involve the academic community in research projects regarding software development problems and challenges.

- DoD should prohibit the acquisition of software applications that do not have Ada bindings.

- Emphasize the rigorous development of Ada 9X, Appendix M, "Interface to Other Languages".

4.0 QUESTION #3: What can we do to increase Ada's appeal, adoption, and popularity within your software community?

FINDINGS

The following findings were the result of participant briefings:

- The Ada community is a close knit and frequently weapons system oriented group. They currently don't communicate well with the MIS community.

- Ada is not highly supported by the DoD acquisition community.

- Acquisition typically does not concern itself with long term support issues.

- Within DoD, "C" (and its variants) is almost a dominant language for research, prototyping, and even many new systems.

- The perception, outside and inside DoD, is that "playing Ada" is an all or nothing game -- variants are not permitted; waivers are often the path of least perceived "cost"; the evolutionary process is very slow and without incentives.

- The perception is that Ada tools lack the modern capabilities of other language tools.

- Program Managers and PEOs are not motivated to consider the long term (maintenance) benefits of Ada.

CONCLUSIONS

Based on the session findings, the following conclusions were made by the panelists

- DoD must package and communicate Ada 9X as the "Now" language.

- DoD's Ada use policy and acquisition methods should dovetail with the way industry wants to sell Ada tools.

- The acquisition process is not well defined or is out-of-date.

- DoD commitment to Ada appears weak.

- The DoD mandate and commitment is not well communicated or defined both inside and outside of DoD.

- Ada tools are not popular or widely used.

- There is a need for quality, portable Ada tools across multiple platforms and environments.

RECOMMENDATIONS

The following recommendations were derived from the findings and conclusions listed above:

- DoD must market Ada 9X more strongly than is currently being done.

- DoD Should ensure Ada compilers and libraries are included in "umbrella" hardware and software procurements.
- DoD should create a formal software engineering education and training program for acquisition personnel.
- Program management plans must consider long term system goals and program management should be graded against their criteria.
- DoD should establish central control over mandates and waivers, and charge that entity with enforcement of and consistency among the Ada rules.
- DoD should inform colleges and universities of Ada tool availability and distribution sources.
- DoD should ensure that quality Ada tools are available to adequately support the software life cycle process.
- DoD should ensure that Ada's support environment is comparable to other development environments.
- DoD should ensure that a rich set of tools and reusable components exist for Ada 9X when it is released.

5.0 PANEL

Panel members were selected to represent various facets of the industrial community.

	Name	Organization
Panel Chair:	Dr Kurt Fischer	OASD(C3I), ODASD(IM)
Panel Facilitators:	Mr. Fred Moxley	Defense Information Systems Agency
	Mr. Anthony Shumskas	Logicon
	Mr. Fred Baldwin	Intermetrics, Incorporated
Panel Members:	Mr. John Foreman	ARPA
	Mr. Alden Wynn	DOE/INEL
	Mr. Thomas Rhodes	NIST/CSL
	Mr. Jack Garman	NASA, Johnson Space Center
	Mr. Jim Hess	DISC4/ARMY

6.0 PARTICIPANTS

Participants were allocated a maximum of 15 minutes to give panel members their comments to questions within the government discipline. The following participants presented briefings during the Ada Dual-Use Workshop (Refer to Section 8.0 of this Appendix, Participant Briefings for copies of presentations). Participants are listed in the order of presentation.

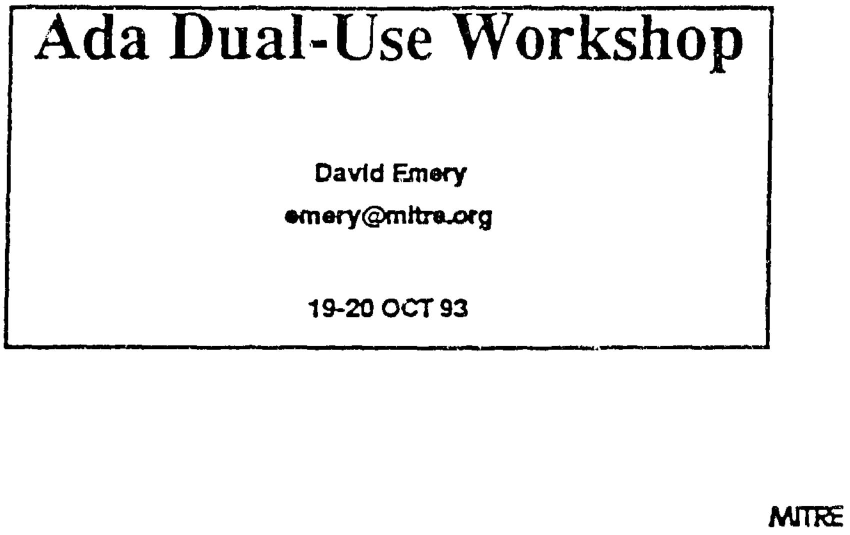
Name	Organization	Briefing
Howes, Norman	Institute for Defense Analyses	Presentation - No Slides
Emery, David	The Mitre Corporation	Presentation - Slides
Siegel, Barry	NCCOSC-NRAD	Presentation - Slides
Reedy, Chris	The Mitre Corporation	Presentation - Slides
Gallagher, Edward	USACECOM SED	Presentation - No Slides
Roby, Clyde	Institute for Defense Analyses	Presentation - Slides
Hook, Audrey	Institute for Defense Analyses	Presentation - Position Paper
Thurmond, George LTC	PEO STAMIS	Presentation - Slides
Depasquale, Gerry Maj	U.S. Marine Corps-MCCTA	Presentation - Slides
Solomond, John	HQ Army Material Cmd	Presentation - Slides
Sodhi, Jag	U.S. Army	Presentation - Slides
Enright, Jim	U.S. Marine Corps-MCTSSA	Presentation - No Slides
McGarity, M. Joan	NAVCOM TELECOM	Presentation - Slides
Kerner, Judy	The Aerospace Corporation	Presentation - Slides
McPherson, Charles	U.S. Army Material Cmd	Presentation - No Slides
Stuart, Antoinette	ASN (RDA)	Presentation - No Slides
Garman, Jack	NASA Johnson Space Ctr	Presentation - Slides
Foreman, John	ARPA	Presentation - No Slides
Druffel, Larry	Software Engineering Institute	Presentation - No Slides
Berg, Richard	Defense Mapping Agency	Presentation - Position Paper
Milton, Donn	IIT Research Institute	Presentation - Position Paper
Tadda, George 1Lt	U.S. Air Force	Presentation - Slides
Wexelblat, Richard, Dr.	Internal Revenue Service	Presentation - No Slides
Colket, Currie	SPAWAR 331	Presentation - No Slides
Langdon, Larry	Census Bureau	Presentation - No Slides

Note: Some of the views expressed were solely the participant's own and not of his/her organization.

7.0 PARTICIPANT BRIEFINGS AND POSITION PAPERS

This section contains a copy of participant briefing materials provided to the Government Panel during the Ada Dual-Use Workshop. In cases where presentation materials were submitted in handwritten form, typed copies were created by the Ada Dual-Use Workshop staff. These copies are noted accordingly. Following the Participant Briefings is a group of Position Papers submitted by individuals who did not make a presentation but wanted their point of view record.

Participant Briefing
David Emery
The Mitre Corporation

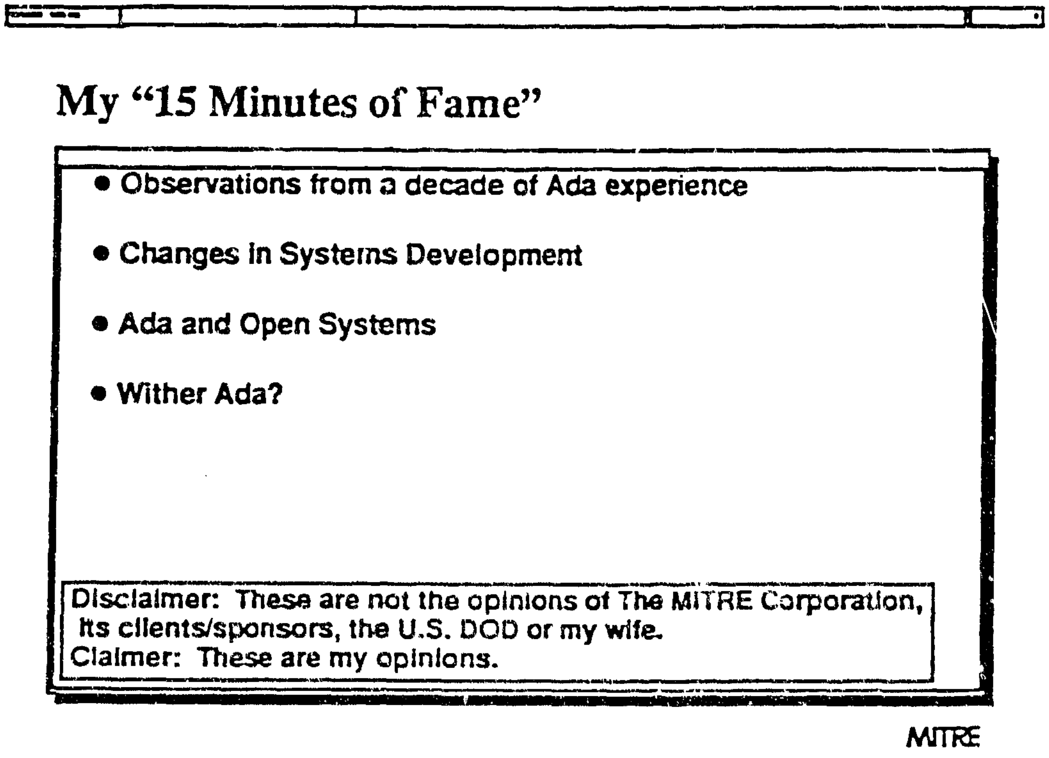


Ada Dual-Use Workshop

David Emery
emery@mitre.org

19-20 OCT 93

MITRE

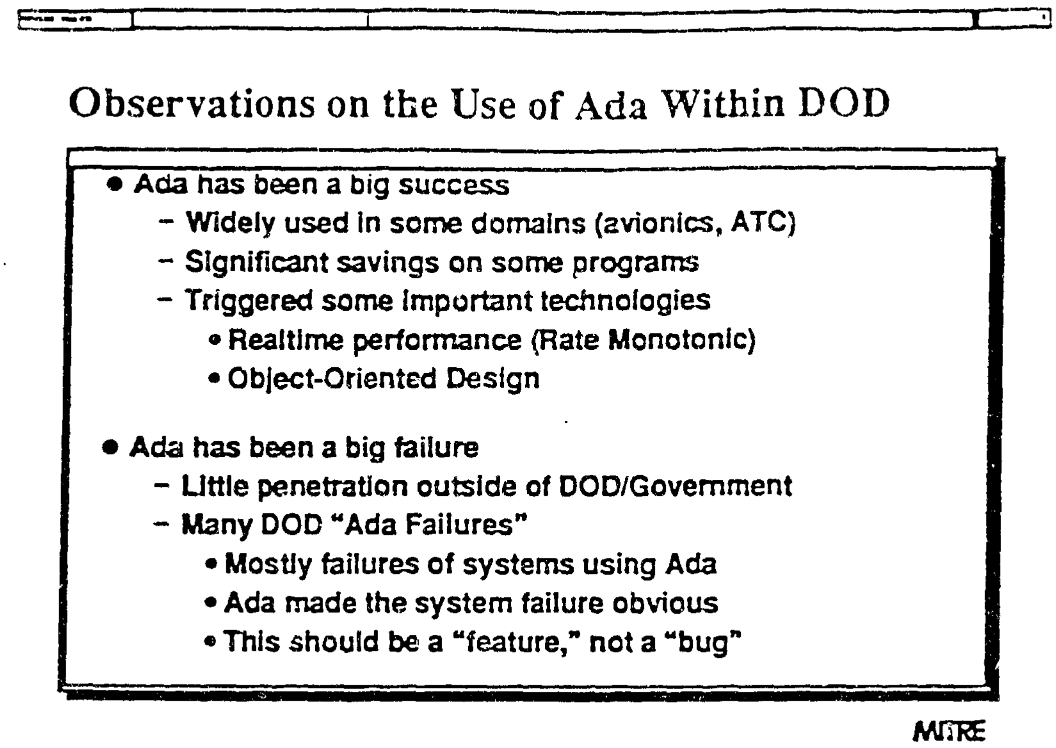


My "15 Minutes of Fame"

- Observations from a decade of Ada experience
- Changes in Systems Development
- Ada and Open Systems
- Wither Ada?

Disclaimer: These are not the opinions of The MITRE Corporation, its clients/sponsors, the U.S. DOD or my wife.
Claimer: These are my opinions.

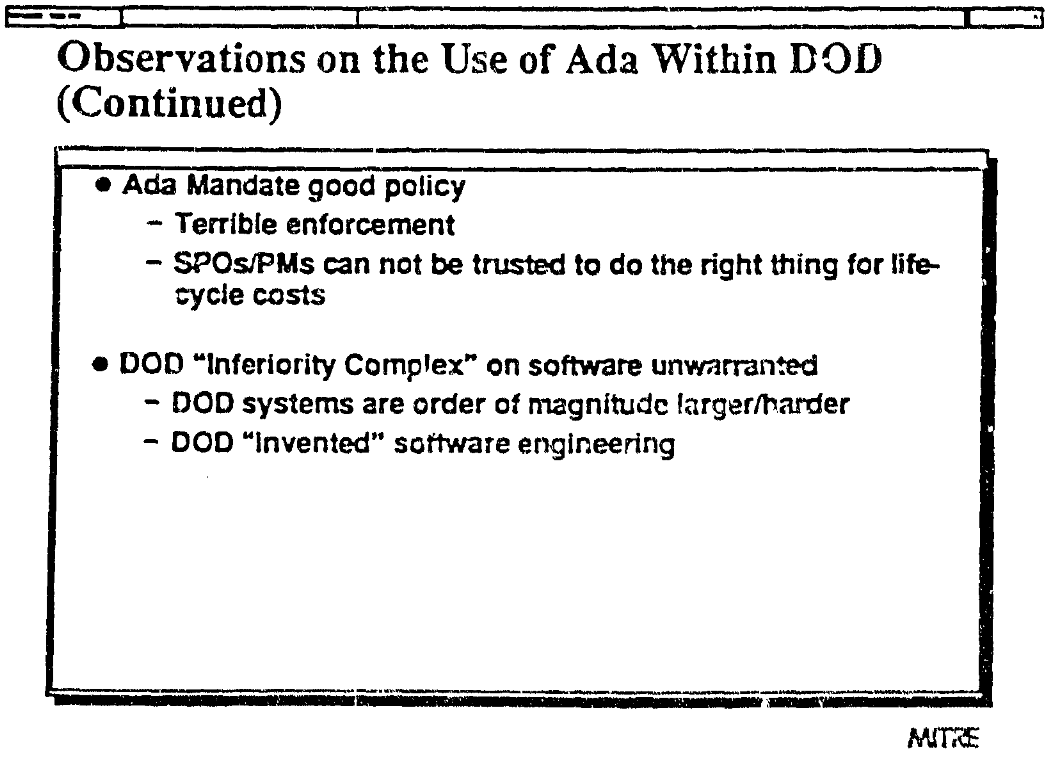
MITRE



Observations on the Use of Ada Within DOD

- Ada has been a big success
  - Widely used in some domains (avionics, ATC)
  - Significant savings on some programs
  - Triggered some important technologies
    - Realtime performance (Rate Monotonic)
    - Object-Oriented Design
- Ada has been a big failure
  - Little penetration outside of DOD/Government
  - Many DOD "Ada Failures"
    - Mostly failures of systems using Ada
    - Ada made the system failure obvious
    - This should be a "feature," not a "bug"

MITRE



Observations on the Use of Ada Within DOD (Continued)

- Ada Mandate good policy
  - Terrible enforcement
  - SPOs/PMs can not be trusted to do the right thing for life-cycle costs
- DOD "Inferiority Complex" on software unwarranted
  - DOD systems are order of magnitude larger/harder
  - DOD "invented" software engineering

MITRE

COBRA DANE Example

- Raytheon estimate: cost $40M to do software in Ada
  - Raytheon had no prior Ada experience, and wanted to use JOVIAL
  - Estimate was $17M MORE to do software in Ada
- Government insisted on Ada
  - Raytheon teamed with TRW for the software
- TRW bid $25M to do software component in Ada
- Software development completed ahead of schedule and within budget (280K SLOC in 24 months, $25M)
- Schedule predicts IOC will occur 3 months early

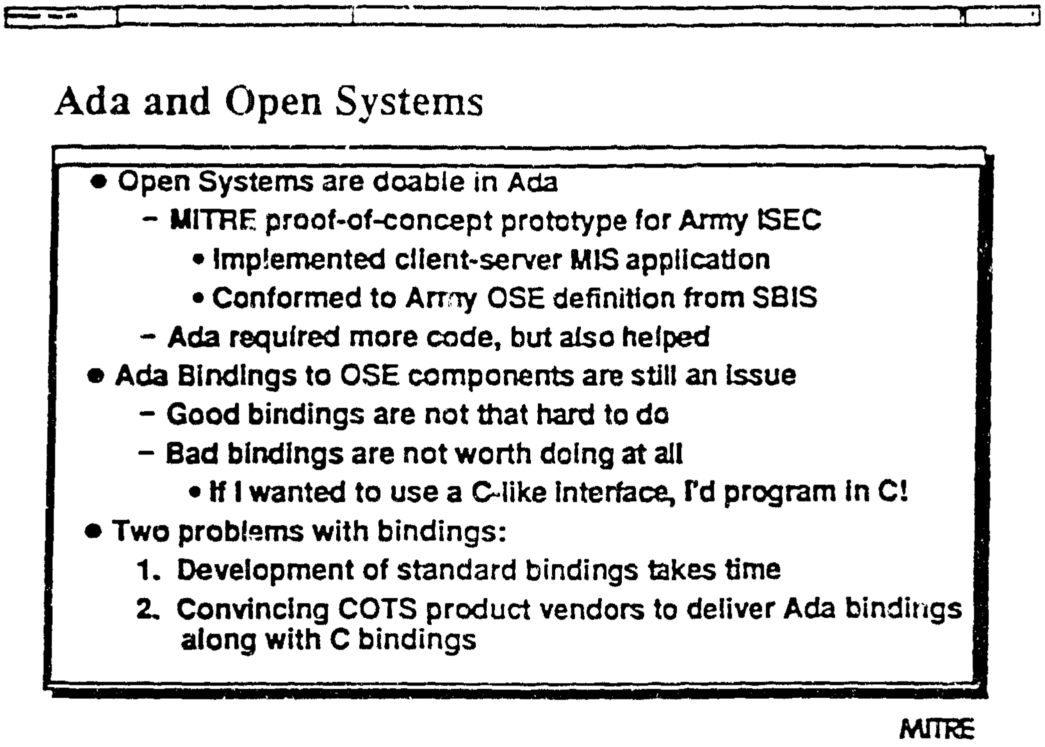
Observation: Ada successes are considered "too good to be true"!

MITRE

Changes in Systems Development

- Technology trends
  - Software as minor component -> Software as major component -> Software IS the system
  - Increases in hardware performance
    - Industry caught up with DOD requiiements
- Software Engineering trends
  - Methodologies
  - CASE
  - Software process
- Now system development is integration, not development
  - Open Systems
  - Reuse
  - Interoperability/Interconnectivity
  - System Architectures

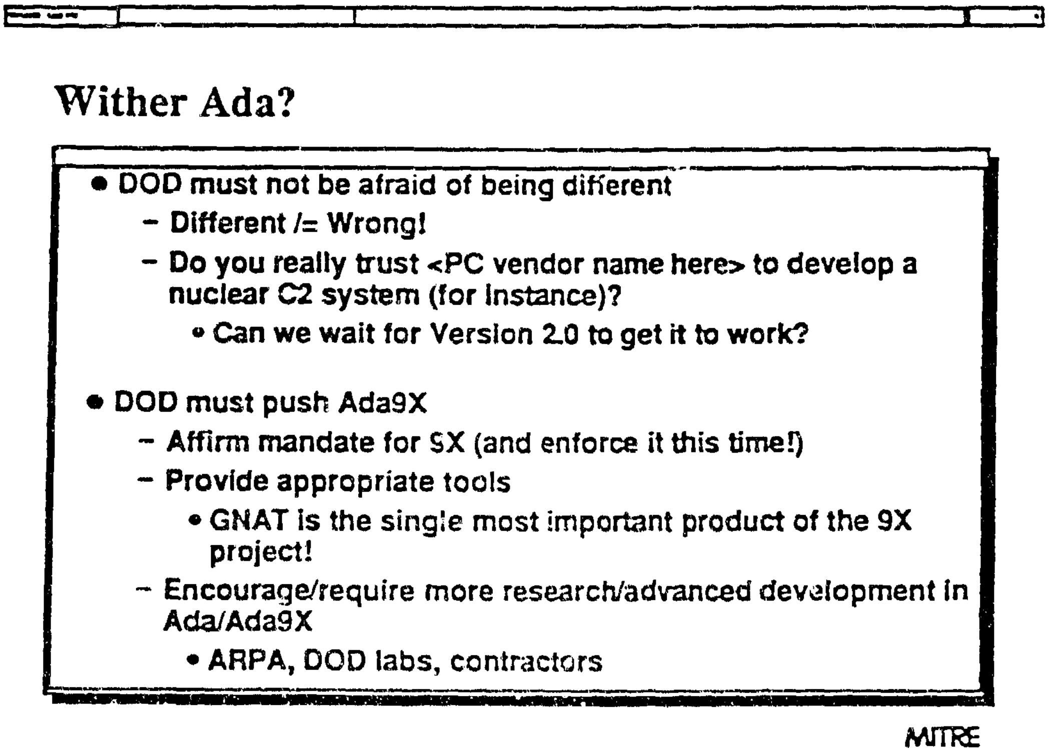
MITRE



Ada and Open Systems

- Open Systems are doable in Ada
  - MITRE proof-of-concept prototype for Army ISEC
    - Implemented client-server MIS application
    - Conformed to Army OSE definition from SBIS
  - Ada required more code, but also helped
- Ada Bindings to OSE components are still an issue
  - Good bindings are not that hard to do
  - Bad bindings are not worth doing at all
    - If I wanted to use a C-like interface, I'd program in C!
- Two problems with bindings:
  1. Development of standard bindings takes time
  2. Convincing COTS product vendors to deliver Ada bindings along with C bindings

MITRE



Wither Ada?

- DOD must not be afraid of being different
  - Different /= Wrong!
  - Do you really trust <PC vendor name here> to develop a nuclear C2 system (for instance)?
    - Can we wait for Version 2.0 to get it to work?
- DOD must push Ada9X
  - Affirm mandate for 9X (and enforce it this time!)
  - Provide appropriate tools
    - GNAT is the single most important product of the 9X project!
  - Encourage/require more research/advanced development in Ada/Ada9X
    - ARPA, DOD labs, contractors

MITRE

Wither Ada?
(Concluded)

- Commercial acceptance of Ada depends on many things
  - Cost (compilers MUST get cheaper/better)
  - Interfaces to other products (bindings, again)
  - Political Correctness/Perception
    - Commercial acceptance of software engineering is real problem
- Ada is quite acceptable as a DOD niche language

"Ada isn't intended for all kinds of software development, ... just software that you intend to use and update for more than one year, software too complex to be written by one person, software that must handle concurrent tasks, software that is readable, portable and reusable, and by the way, software that must be highly reliable and perform well."

"Everything else can be written in C."

Bill Carlson, quoted by Bob Metcalfe, *InfoWorld*, Jul 5 1993

Participant Briefing
Barry Siegel
NCCOSC-NRAD

Naval Command, Control and Ocean Surveillance Center

NRaD

RDT&E Division

BARRY SIEGEL
Code 4123
San Diego, Ca. 92152-5000
OFFICE: (619)-553-4081
FAX: (619)-553-4808
E-MAIL:*siegel@nosc.mil*

WHAT ACTIONS CAN BE TAKE TO MAKE ADA THE PRIMARY TOOL FOR DEVELOPING SOFTWARE SYSTEMS WITH YOUR GOVERNMENT ORGANIZATION?

- CLARIFY THE ADA MANDATE AND WAIVER RULES
- TIGHTEN WAIVER RULES
  - Apply to all projects, including RDT&E 6.1 and 6.2, Nondeliverable SW, etc.
  - Unifomity across services
- EDUCATE PROGRAM MANAGERS ...
  DIRECTIVE MUST COME FROM THE TOP
- PROMOTE THE COHABITATION OF ADA AND C/C++ IN EXISTING PROJECTS
  - There are no technological barrirors to mixing old modules written in C or C++ with new modules written in Ada
- REQUIRE ADA ENVIRONMENTS WITH BLANKET HARDWARE PROCUREMENTS (e.g., Navy TAC-4)

WHAT CAN WE DO TO INCREASE ADA'S APPEAL, ADOPTION, AND POPULARITY WITHIN YOUR SOFTWARE COMMUNITY?

- PROMOTE ADA 9X AS THE "NOW" LANGUAGE (e.g., OOP) AND ACCELERATE ADOPTION
- IMPROVE THE ADA DEVELOPMENT ENVIRONMENT
  - GUI BUILDERS, X-WINDOWS FRONT END
- LOWER THE COST OF ADA COMPILERS; SW MAINTENANCE, TRAINING AND THE DEVELOPMENT ENVIRONMENT
- DEVELOP AND PROMOTE BINDINGS TO POPULAR COMMERCIAL STANDARDS (e.g., Windows, OLE-2)

EXAMPLE: COST and AVAILABILITY OF COMPILERS AND TOOLS

	C++	ADA
DOS	$80	$200
WINDOWS	$200	$1000
UNIX (per seat)	$2500	$7500
MOTIF BINDINGS	$200	$2500
AVE # OF COMPILERS (per platform)		
	5-6	2-3

WHAT INVESTMENTS CAN WE MAKE TO INCREASE THE COMMERCIAL USE OF ADA-1

- **INVESTMENTS THAT INCREASE THE GENERAL AVAILABILTY OF ADA COMPILERS**
  - But compeition unlikely in high investment, low volume, high risk technical software markets
  - Only competitive market is PC-Based C++ Compilers
- **INVESTMENTS THAT LOWER THE COST OR INCREASE THE PRODUCTIVITY OF ADA IN SOFTWARE DEVELOPMENT**
  - Ada will be adopted only when there is competitive advantage
  - Capital (e.g., compilers, reuse libraries, etc.) and Labor (Programmmers) are substitutable.

WHAT INVESTMENTS CAN WE MAKE TO INCREASE THE COMMERCIAL USE OF ADA-2?

- INCREASE THE SUPPLY OF PROGRAMMERS
  - University Programs in Ada
- INCREASE THE SUPPLY OF "ADA" CAPITAL
  - Ada Technology Insertion Programs such as Windows/NT GUI Builder
  - Reuse Libraries
- INCREASE ADA PRODUCTIVITY
  - University Programs in Ada
  - Low-Cost Reuse and Reengineering Tools
  - Reduce barriors to SW reuse (i.e., legal barriers, Essential Technologies and Systems or EPITS)
- MARKET FORCES WILL DETERMINE THE FATE OF COMMERICAL USE OF ADA

PROTECTIVE MARKINGS CANCELLED
WHEN INCLOSURES ARE WITHDRAWN

DEPARTMENT OF THE ARMY
PROJECT MANAGER
FIELD ARTILLERY TACTICAL DATA SYSTEMS (FATDS)
PROGRAM EXECUTIVE OFFICE-COMMAND AND CONTROL SYSTEMS
FORT MONMOUTH, NEW JERSEY 07703-5404

SFAE-CC-FS-A (70-17b)

15 JUN 1993

MEMORANDUM FOR SEE DISTRIBUTION

SUBJECT: Advanced Field Artillery Tactical Data System (AFATDS) Acquisition System Program Protection Plan Explanation and Protection of Essential Program Information, Technologies or Systems (EPITS)

1. Assistant Secretary of the Army Research, Development and Acquisition (ASARDA) has recently implemented the Army Acquisition Systems Protection Program. The amount of time between the fielding of a new U.S. weapon system and the enemy fielding a counter system or copy has been steadily eroding to the point where enemy counters are fielded almost simultaneously or even before the U.S. weapon system fieldings. In addition, friendly allies have been selling copies of U.S. weapon systems based on key U.S. technologies thereby diminishing our market share and decreasing our economic advantage.

2. As a result of the above, DoD has mandated that all Acquisition Category IC and ID programs must have an Acquisition Program Protection Plan (Acq P3) updated at each milestone. The Acq P3 identifies critical or unique technologies of a system and states how they will be protected. The critical or unique technologies are called EPITS.

3. The EPITS are not necessarily classified but since they are what "makes it tick" they must be protected. The Acq P3 provides guidance on how the EPITS are to be protected. Since the AFATDS Acq P3 will not be completed for a few months, this memo will provide interim, cursory guidance for protection of the AFATDS EPITS. Since software has unique properties making it difficult to separate EPITS from the entire AFATDS software program, the following are instructions for protecting the entire AFATDS program and consequently the AFATDS EPITS:

a. Access to the AFATDS source code should be limited to personnel who have direct involvement with AFATDS or those who have been authorized access by the PM or DPM AFATDS. This should include remote on-line access.

b. Do not allow copies of either part or the entire AFATDS source code to be made or taken from the area.

PROTECTIVE MARKINGS CANCELLED
WHEN ENCLOSURES ARE WITHDRAWN

SFAE-CC-FS-A (70-17b) 15 Jun 1993

SUBJECT: Advanced Field Artillery Tactical Data System (AFATDS) Acquisition System Program Protection Plan Explanation and Protection of Essential Program Information, Technologies or Systems (EPITS)

c. Copies of the source code should not be distributed without authorization of the PM or DPM AFATDS.

d. When copies of software are distributed after proper authorization, keep a log of who has received copies of the software. Ensure by signature of the recipient acknowledgment, understanding and agreement to abide by the above requirements. (This office will provide a letter to be used to a receipt.)

e. The EPITS (Encl) should be handled as FOUO and not be distributed or discussed as public information but released only to the minimum number of involved individuals.

4. The POC for this action is Mr. Nicholas Keselowsky, at DSN 992-0336 or Commercial (908) 532-0336.

Encl

For Arthur Santo Donato
STEVEN W. BOUTELLE
COL, SC
Project Manager

DISTRIBUTION:
Director, TRADOC System Manager, Fire Support Command, Control & Communications, ATTN: ATSF-TSM-C3A, Ft Sill, OK 73503-5600
Director, Center for Software Engineering, Fire Support Software Engineering, ATTN: AMSEL-RD-SE-SY-FS-S, Ft Sill, OK 73503-5600
PM CHS, ATTN: SFAE-CC-CHS/SFAE-CC-CS, Ft Monmouth, NJ 07703-5404
Chief, Fire Support, Bldg 461, ATTN: AMSEL-RD-SE-SY-FS, Ft Sill, OK 73503-6600
Marine Corps Systems Command, ATTN: C2G, Quantico, VA 22134-5010

FOR OFFICIAL USE ONLY

SFAE-CC-FS-A

MEMORANDUM FOR Commander, Fort Monmouth MI Detachment,
MI Battalion (CI) (S), ATTN: IAGPA-B-MO,
Fort Monmouth, NJ 07703

SUBJECT: AFATDS Essential Program Information, Technologies or Systems (EPITS)

1. Following are the AFATDS EPITS. It should be noted at this time that, since the Acquisition Protection Program is new, the PM reserves the right to modify this list as our understanding of the Protection Program matures.

2. AFATDS EPITS :

A. Rule Sets/Algorithms

These are guidance packages which determines how AFATDS will react based upon input parameters. Knowledge of these rule sets would enable an enemy to force AFATDS to react in a manner beneficial to an enemy.

Specifically:

1. Attack System Analysis - Depending on target and commander's guidance, enables AFATDS to select unit and munition.
2. Weapons System Selection - Enables AFATDS to select the optimal weapon system available to successfully complete a fire mission with the minimum asset expenditure.
3. Determination of Unit Capability/Availability - Enables AFATDS to assertain the status of a particular unit with regard to quantities of munitions, manpower availability and unit commander's opinion of unit's readiness among other factors. AFATDS will select/reject a particular unit based on this status.
4. Volume of Fire - Calculates the number of volleys required to achieve the desired effect. Under this two specific algorithms should be protected :
    a. SMART
    b. SUPER QUICKIE II

B. Communications:

1. TACFIRE/LTACFIRE serialization and authentication algorthim.
Knowledge of this would enable an enemy to mimic and "spoof" AFATDS with false messages proportedly from TACFIRE/LTACFIRE.

2. The AFATDS applications software contains information/algorthims concerning finite details of the communications architecture. The applications software that deal with network access structures and COMSEC key flags should be EPITS. Such information would facilitate where and when jamming would be mosr effective.

3. The TCIM (Tactical Communications Interface Module) is a smart modem used with AFATDS which enables CHS computers to connect with Army tactical communications. The AFATDS applications software "loads" the control software and communication protocols into the TCIM. The TCIM control software and the TCIM communications protocols should be protected as EPITS. Knowledge of these would facilitate an enemy intercepting legitimate messages and sending false messages.

C. JMEMs Data :

The JMEM (Joint Munitions Effects Manual) effect data is classified as SECRET and is handled as such. The AFATDS Applications software that handles the effects tables data should be an EPITS.

3. Point of contact is Mr. Nicholas Keselowsky, Comm. (908)532-0336, DSN 992-0336.

STEVEN W. BOUTELLE
COL, SC
Project Manger

Participant Briefing

Christopher Reedy
The Mitre Corporation

BackGround

- Based upon prior work experience
  - DBMS development
  - Database applications
  - Artificial Intelligence
- These are my personal opinions, not those of my employer

"Technical"

- Need Ada 9X desperately
  - Too many missing features in Ada 83, e.g. inheritance
- Bindings
  - UNIX, X, SQL
  - Needed five years ago
- "Visual Ada"
  - Needed - Create acceptance of Ada on personal computers
  - Includes large class libraries and visual programming

Language

- Current “business” applications are being built using 4GL/Visual Programming
- 3GLs (e.g. C/C++ and Ada) are being relegated to “systems” programming and “extensions”
- Ada must be usable as an extensions language
  - eliminate (or minimize) run-time library requirement
  - callable from other languages without an Ada “main”

“Standards”

- Most of computing community proceeds ahead of standards
  - C++ de facto standard, not yet official standard
- Ada community lags behind standards development
  - Must standardize before software/tools become available
- Need to encourage innovation with Ada
  - E.g., accept language “extensions”

Participant Briefing
Clyde Roby
Institute for Defense Analyses

Ada Dual-Use Workshop

19-20 October 1993
Vienna, VA

Clyde Roby
Institute for Defense Analyses
1801 N. Beauregard Street
Alexandria, VA 22311
(703) 845-6666
FAX: (703) 845-6678
E-mail: Roby@IDA.Org

With respect to Ada Bindings...

- What actions can we take to make Ada the primary tool for developing software systems within your government organization?
- What investments can we make to increase the commercial use of Ada?
- What can we do to increase Ada's appeal, adoption, and popularity within your software community?

Ada bindings to ...?

- To defacto standards
- To COTS "standard" products
- To widely-used national and international standards

Examples:
X-Windows, Microsoft Windows,
Kermit, SQL, PCTE, e-mail

Why worry about Ada bindings?

- So that there are not multiple Ada bindings to a standard or to a product — X, for example
- So that there is some consistency to the Ada bindings ("thin" versus "thick" bindings) — early POSIX

Why a single Ada binding?

- So that implementors have only one binding to implement — not several
- So that tool vendors can develop their tools once, then port them to different implementations of the Ada binding
- So that end-users can develop their applications once
- So that training for Ada development will be more uniform and thus less costly

Ada Bindings

- Investigate Ada bindings to popular products and standards, such as those mentioned earlier
- If multiple Ada bindings exist, narrow the choice down to (hopefully) one
- If no Ada bindings exist, encourage cooperation among interested parties to define an Ada binding
- Once the Ada binding exists, publish the Ada package specifications widely and publicly, and encourage the market to supply package bodies
- Perhaps, in some cases, standardize the Ada package specifications

Ada Bindings

- Find out what people would like Ada bindings to
- Go beyond the Ada community and get other parties interested in the development of Ada bindings — commercial vendors, standards groups, ...
- Increase the number of other products/systems that are accessible to Ada programs
- Many vendors now deliver products with other language bindings, can we ask them to deliver Ada language bindings, too?
- Increase coordination with standards groups to influence their development so that Ada bindings can readily be defined
- Promote Ada bindings

Ada Bindings

- Investigate the "right" way to develop Ada bindings by looking at:
  - Ada binding technology: what is the best way to define an Ada binding?
  - How to go about developing an Ada binding
  - How to move from another language binding to Ada
  - "Thin" versus "Thick" Ada bindings
- Make existing Ada bindings better, expecially for Ada9X
- Try to affect products during the development stage
- Try to affect standards during the standardization process, not after they have become standards
- Publicize who is producting/providing Ada bindings and what are they interfacing to

Ada bindings are good for ...

- Academia
- Government
- Industry
- Hardware and Software Vendors

Academia

Having Ada bindings to the popular products and tools, especially on PCs and workstations, will increase interest in Ada by instructors, thus making it a primary teacning language at major universities; this will also increase the pool of entry level personnel with Ada skills.

Government

Having Ada bindings to existing program building blocks will increase the adoption and popularity within DoD and other government organizations, thus making Ada a primary tool for developing large software systems.

Industry

Having implementations of Ada bindings ubiquitous and inexpensive can help Ada become the preferred language for developing software, increasing its appeal and popularity within industry.

Hardware and Software vendors

Having Ada bindings will help to increase the market for Ada products and services beyond the defense community, thus investing in the development and possibly the standardization of Ada bindings will increase Ada's commercial use.

Participant Briefing
LTC George Thurmond, II, US Army
PEO STAMIS

Department of Army

PEO STAMIS

LTC George E. Thurmond II

Acting Director
Software Engineering & Integration Dir

Ada Dual-Use Workshop (Govt)
19 Oct 93

Strategy

Government Questions Inextricably Linked

- Establish Clear Policy & Direction
- Build Foundation
- Reward Investment and Responsiveness
- Share Technology
- Promote Successes

PEO STAMIS

Actions

Acquire Ada Software Development Support Through Defined (OSE), Disciplined and Measurable (Automated Tool Set) Process

Energize Vendors of CASE Products to Exercise Unique Ada 9X Language Features

Continue Culture Change Through Education, Up Front Planning, Access to Resources

PEO STAMIS

Investments

Universities - Partnership with Industry to Promote SW Engineering Using Ada

Create Incentives for Tech Proposals Demonstrating Ada Expertise, Reuse Capability and Historical Tech Transfer Across Projects

Distribution of Public Domain Software, Program Sponsorship Such as with Historically Black Colleges

Ex: AdaSAGE Engineering and Management Group

Acceptance

DoD Mandate

Demonstrated Savings Associated w/ Development & PDSS Costs

Cultural/Emotional Issues:

- Easy to Use
- Productive
- Affordable

Participant Briefing
Maj Gerry Depasquale, US Marine Corps
US Marine Corps - MCTSSA

Marine Corps AdaSAGE Experiences

Presented by

Major Gerald A. DePasquale
Marine Corps Computer and Telecommunications Activity
Quantico, Virginia

Request

$1 Million Annually to Support AdaSAGE

[Commitment will pay large dividends.]

Lesson 1

Marines love AdaSAGE.

[AdaSAGE is a Marine Corps standard development tool.]

Lesson 2

AdaSAGE is good - Ada is bad.

[To use AdaSAGE effectively,
introductory level Ada training is required.]

Lesson 3

AdaSAGE means

"you get a meritorious service medal"

[Using AdaSAGE, new Ada programmers are productive in six weeks vice six months.]

Lesson 4

AdaSAGE means

"you get a vacation"

[AdaSAGE reduces our development time by 50%.]

Lesson 5

AdaSAGE means

"you get mo money, when you lie about your effort"

[AdaSAGE provides 55% of our applications.]

Lesson 6

AdaSAGE means

"commercial DBMS vendors must get serious about Ada"

[AdaSAGE often outperforms commercial DBMS products.]

Lesson 7

AdaSAGE is

"a deadly weapon in the hands of a high school graduate".

[A high school graduate
+ one month of Ada training
+ two weeks of AdaSAGE training
= productive programmer.]

Lesson 8

AdaSAGE Video Training means

"you don't have to train that stupid bastard"

[Beginning AdaSAGE Video Training Materials
provide effective refresher and introductory education.]

Lesson 9

AdaSAGE Video Training also means

"you catch up on your sleep"

[You must stay awake when viewing the Advanced AdaSAGE Video Training Materials to glean valuable advanced knowledge.]

Lesson 10

AdaSAGE means

"you can let the students do all the work"

[AdaSAGE can be used as an aid
to teach software engineering teams.]

Lesson 11

Enhanced AdaSAGE means

"everyone will love you"

[Marine programmers have rewritten applications developed back in late 1980s, adding new functionality made possible by AdaSAGE enhancements, while simultaneously reducing SLOC, increasing execution speed, reducing executable size, and improving maintainability.]

Lesson 12

AdaSAGE Users Group means

"if you have money, you have a friend"

[Enrollment in the AdaSAGE users group is worth the investment. Technical difficulties, wrestled with for days, have been solved over a single phone call.]

Lesson 13

AdaSAGE management means

"you own the system"

[Enhancements requested by the user community get completed.]

Lesson 14

AdaSAGE means

"damn those compiler vendors"

[The Corps has purchased compilers explicitly for use with AdaSAGE only to find out that the new compiler release is incompatible with AdaSAGE.]

Lesson 15

Badmouthing AdaSAGE

"can get you killed"

[Captain Hernandez, IRMD Albany, challenges all detractors to talk to him before claiming AdaSAGE can not meet their general purpose data base management programming needs.]

Participant Briefing
John Solomond
HQ US Army Materiel Command

ADA SUMMIT

19-20 OCT 93

JOHN P. SOLOMOND

ARMY MATERIEL COMMAND

PRESENTING OBSERVATIONS OF ADA BOARD:
DEC 92, APR 93

PERCEPTIONS AND REALITY USING Ada

Major categories

A. Economic/ business issues
B. Education/ research issues
C. Systems integration problems
D. DoD Ada mandate, acquisition policies
E. Technical issues

ECONOMIC/ BUSINESS ISSUE: EXPENSE

Issues:

Ada compilers too expensive, large, and complex; development costs too high

Ada compilers are large and require more resources than compilers for traditional languages, e.g. "C"

Ada users develop on large projects that demand additional expensive tools to achieve optimum productivity

Operating costs to develop a first system are higher for Ada, particularly as a result of the lack of trained personnel at all levels. Situation improves drastically, however, with increased training of personnel

RESPONSE

Compiler vendors will indeed have to invest large amounts of resources

However, comparable transition to C++ from C will be large as well. Retooling and retraining for this transition is expensive for firms using C on large projects.

Object oriented features of Ada 9X eliminate claims of superiority by C++

Maintain standardization effort using 9X via the international standardization route

ECONOMIC/ BUSINESS ISSUES

Issue	Response
Reuse of Ada programs is not meeting expectations The Japanese have publicized success with significant levels of code reuse, but the US has not	American firms consider productivity gains through Ada reuse a competitive advantage - do not want to highlight advantages of Ada

ECONOMIC/ BUSINESS ISSUE

Issues	Response
Compiler vendors are not marketing Ada sufficiently. This is claimed in relation to marketing of C++.	Size of Ada market is small relative to C++. Ada vendors tend to concentrate in small niche areas of the market. Greater flexibility and technical interest associated with Ada 9X should offset this trend.

EDUCATION/RESEARCH ISSUES

Issue	Response
Ada should be used by the research community	Researchers have investments in tools that facilitate use of other languages Lack of incentives to conduct research in "new" language, especially when already comfortable using C or FORTRAN
Financial incentives for Universities to use Ada	AJPO/DARPA initiative for Ada introduction via software-engineering courses
Learning curve for first use in education	Professors concerned about steep learning curve to bring into curricula Easy to learn - more difficult to teach

EDUCATION/RESEARCH ISSUES

Issue	Response
Ability to deploy limited resources to optimize introduction of Ada into education process	Large startup costs of several million dollars per year Ada requires more powerful or at least as large amount of memory to populate laboratory
Transition to Ada technology from research to Ada applications	Research tends to be done in other languages Principal investigators on DoD contractors need to have incentives for doing research in Ada

EDUCATION/RESEARCH ISSUES

Issue	Response
Must transition from prototypes into production using Ada. They currently tend to be written in C/ C++/Lisp	Transition from prototype to production seems to be enhanced; language remains the same
Overhead of using Ada is too large for teaching	Designed for large-scale systems; using Ada with small classroom examples sometimes not appropriate

SYSTEM INTEGRATION PROBLEMS

Issue	Response
Ada weak in large-scale systems integration.	Many of the characteristics of Ada, which make it suitable for the development of large systems, make it suitable for the integration of large systems: Ada's encapsulation and modularization facilities improve the reliability and simplify the maintenance of large systems, regardless of whether they are implemented using development or integration strategies. Ada's implicit enforcement of sound software-engineering principles like strong typing have the effect of ensuring that the interface assumptions of the subsystems are not violated.

SYSTEM-INTEGRATION PROBLEMS

Issue	Response
Lack of Ada bindings to key "open systems" forces choices between the language portability provided by Ada and the platform portability provided by open systems	Some open systems standards have recently provided Ada bindings, e.g. SQL and POSIX.

DoD Ada MANDATE- ACQUISITION POLICIES

Issue	Response
Procurement practices incompatible with Ada benefits	Financial arguments for using Ada are based on lifecycle cost savings, while procurement practices emphasize front end of the lifecycle. Benefits of Standardization and Logistics issues were primary criteria for moving to a standard

DoD Ada MANDATE- ACQUISITION POLICIES

Issue	Response
Scope of mandate perceived as being too broad	Original application domain was for embedded systems No single language is suitable for all conceivable application domains UK ministry of Defense does not "mandate" any language, but requires validation and evaluation of compilers to suit needs of project office -- Result: very few projects bid without specifying the Ada language. UK Defense Industry largely supports the language Require validation of other non-Ada languages as well

TECHNICAL ISSUES

Issue	Response
Ada is inferior to C/ C++	Research proves otherwise: Ada vs C++ Ada 9X vs C++
Ada not best technical choice for all problems	Ada has proven successful in AI and real-time applications Support for distributed/ parallel processing untested
Bindings slow in coming	ATIP is helping

RECOMMENDATIONS

(PERSONAL-NOT BASED ON OFFICIAL BOARD CONSENSUS)

- DO NOT DROP MANDATE
  - RELAX IT FOR EXPLORATORY WORK & PROTOTYPES
    - SUBSETS
  - NON-VALIDATED COMPILERS
    - EDUCATIONAL INSTITUTIONS
    - RESEARCH INSTITUTIONS
- MODIFY ACQUISITION GUIDELINES
  - LIFE CYCLE COSTS VICE ACQUISITION COSTS
- FINANCIAL INCENTIVES TO GOVERNMENT USERS RELUCTANT TO CONVERT TO ADA, E.G. ARMY DEPOT

Participant Briefing
Jag Sodhi
US Army

(Presentation materials were submitted in handwritten form, a typed copy was created by the Ada Dual-Use Workshop staff.)

JAG SODHI
US ARMY, CECOM, RDEL

SED, FIRE SUPPORT SOFTWARE
ENGINEER
FORT SILL, OK

•

- Fire Support/Field Artillery
- 32 Systems
- Ada
- Mandate

Question 1: Government

- Mandate
- Education and Training
- Incentive
- Tools
  - Compiler
  - Ada Bindings
- Software Reuse
  - Pilot Projects
- 20% Funding
- Ada 9X

Question 2: Commercial

- Investment
- Education and Training
- Software Engineering
- Fill Up Gap (Software Requirements > Design > Ada)
- Mapping

Question 3: Ada's Appeal

- Simple
- KISS
- Advertisement
- Qualified Professionals
- Success
- Textbooks
- Lessons Learned
- Sharing

Question 3 (cont): Ada's Appeal

- Compilers (Cost Effective)
- Funding for Universities
- Follow Marketing Rules
  - Easy to Follow
  - Workable
  - Cost
  - Time
  - Quality

Participant Briefing
M. Joan McGarity
NAVCOM TELECOM

Ada Dual Use Workshop

Government Track

Ms. Joan McGarity
NAVCOMTELCOM

19 - 20 October 1993

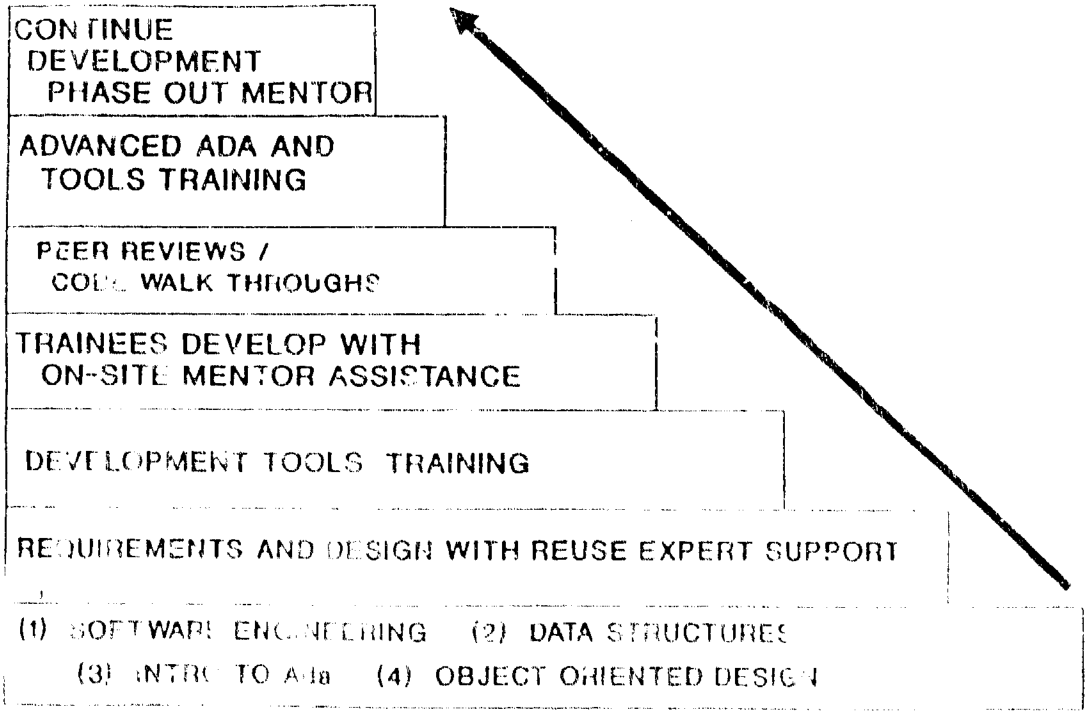
CONTEXT

- Software Engineering
- Economics
- Culture
- Immature Tools

Ada as the primary development tool

- Provide technical support
- Effective, easy access to training
- Enforce the mandate
- Fee-for-service responds to customer
- Access to Reuse Repositories
- Effective tools

Ada TRAINING/DEVELOPMENT PROCESS



CONTINUE DEVELOPMENT PHASE OUT MENTOR

ADVANCED ADA AND TOOLS TRAINING

PEER REVIEWS / CODE WALK THROUGHS

TRAINEES DEVELOP WITH ON-SITE MENTOR ASSISTANCE

DEVELOPMENT TOOLS TRAINING

REQUIREMENTS AND DESIGN WITH REUSE EXPERT SUPPORT

(1) SOFTWARE ENGINEERING (2) DATA STRUCTURES
(3) INTRO TO Ada (4) OBJECT ORIENTED DESIGN

Increasing Commercial Use of Ada

- Tie Ada to business applications
- Lower cost, more powerful tools
- Leverage off existing government-owned tool, AdaSAGE
- Provide quality, inexpensive, fun training packages

Increase Appeal, Adoption & Popularity

- Educate our customers
- Quality tools
- Publize success
- Keep AdaSAGE current with standards
- Be positively aggressive
- Provide bindings

DRAFT

5230
Ser N91/

From: Commanding Officer, Naval Computer and Telecommunications
Area Master Station LANT
To: Commander, Naval Computer and Telecommunications Command,
4401 Massachusetts Avenue, N.W., Building 19, Washington,
DC 20394-5000

Subj: INITIAL PLANNING ESTIMATE FOR NAVY ADA TECHNICAL SUPPORT PROJECT FOR FY94

Ref: (a) NAVDACINST 5320.1C

1. In accordance with reference (a), an Initial Planning Estimate for continuing support of the Navy Ada Technical Support Project (b) is as follows:

a. Project Number: N912ADA

b. Project Title: Navy Ada Technical Support

c Date Project Received: 18 September 1992

d. Responsible Developer: NCTAMS LANT

e. Planning Estimate Responsibility: Mr. T. Colin Pratt, N91, Autovon 564-3873 or Commercial (804) 444-3873

f. Project Responsibility: Mr. T. Colin Pratt, N91, Autovon 564-3873 or Commercial (804) 444-3873

g. Significant Dates:

Proposed Starting Date:	01 October 1993
Estimated Completion Date:	30 September 1994

h. Projected Costs:

Labor: (3,516 hours @ $37.15 per hour)	$130,643
Travel and Misc.:	$ 25,000
Total	$155,643

i. Description of Services:

Provide software engineering and Ada technical leadership to the COMNAVCOMTELCOM activities and the Ada technical community at large. Represent COMNAVCOMTELCOM at significant technical reviews which impact or provide direction for the growth of Ada.

DRAFT

Subj: INITIAL PLANNING ESTIMATE FOR NAVY ADA TECHNICAL SUPPORT PROJECT

j. Deliverables:

(1) Attend Ada conferences and workshops as directed and funded by COMNAVCOMTELCOM.

(2) Review and make recommendations on Ada tools and software engineering techniq .

(3) Conduct meetings and/or seminars for the purpose of communicating Navy requirements to Ada vendors.

(4) Provide COMNAVCOMTELCOM with a monthly status report.

(5) Write articles to be published in Navy and Ada technical community publications.

(6) Maintain and upgrade information provided to the Ada community on the Ada Technical Bulletin Board, to include sections on the following:

(a) AdaSAGE

(b) Reuse Program

(c) Complaints

(7) Write Ada bindings, where not provided by the vendor community, to attract users on the microcomputer platform.

(8) Publish COMNAVCOMTELCOM's development activities successes with Ada in the management information system environment.

(9) Provide a quarterly report to COMNAVCOMTELCOM on services, results and significant information of COMNAVCOMTELCOM's development activities.

(10) Provide recommendations for an Ada Training Methodology and the required skills necessary at different levels of training.

(11) Provide a Plan of Action and Milestones.

2. If you agree with this Initial Planning Estimate, please furnish formal acceptance and funding.

J. M. SHUMATE
By direction

Participant Briefing
Judy Kerner
The Aerospace Corporation

Ada Issues at the Air Force Space and Missile Systems Center

Aerospace Corporation Perspective

J. Kerner
Engineering Specialist

"Ada Problems" at SMC

- Most contractors (and Program Offices) do not have the necessary depth and breadth of expertise in Ada
- Many programs are using non-Ada COTS products and legacy software that are often difficult to integrate with Ada (COTS/Ada bindings are incomplete and/or late, data formats are difficult to map, etc.)
- Ada compilers (and development environments in general) are still immature for many popular platforms
- Some contractors are labeling proprietary software (not written in Ada) as "COTS" or "NDI" just to circumvent the Ada mandate, resulting in less government visibility into the products, more maintenance problems

What actions can we take to make Ada the primary tool for developing software systems within your government organization?

Technical Issues

- Get commercial software product vendors to develop COTS products in Ada, or at least provide **timely, fully functional** support for Ada interfaces
- Develop standard packages/paradigms for interfacing with legacy code in other languages
- Ensure that support environments of comparable **quality** and **cost** to those for other languages are available on all appropriate platforms
- Support compilation technology and the compiler market so that compiled Ada code is not less efficient or larger than "comparable" C or assembly code

What actions can we take to make Ada the primary tool for developing software systems within your government organization?

Management Incentives

- Enforce the Mandate uniformly across DOD with easy exceptions/waivers for interface code
- Encourage/incentivize use of subcontractors who are proficient in Ada
- Support SPOs who want to use Ada with additional funding so they are not shouldering the whole cost of the contractors' learning curve as well as their own
- Increase Ada's appeal and popularity in the contractor/government software community
- Increase the awareness at high levels of the importance of software and software engineering - educate all levels on the impact of technical issues
- Change the acquisition infrastructure so program managers are incentivized for long-term cost effectiveness over short-term cost/schedule performance

What investments can we make to increase the commercial use of Ada?

- Publicize successful commercial applications of Ada, both U.S. and foreign
- Invest in software libraries to make good Ada code widely available (and it should be free!)
- Strengthen government and non-government marketing of Ada
- Get academia (universities, colleges, junior colleges, even high schools!) interested in Ada so there are lots of graduates who know it
  - Market at conferences, publish papers, get academics interested in Ada's features
  - Provide academia with effective environments for free (or very cheap)
  - Encourage the use of a "teaching subset" of Ada, especially as a transition approach from Pascal-based programs

What can we do to increase Ada's appeal, adoption, and popularity within your software community?

- Make sure guidelines are widely available for how to write common paradigms in Ada (more than a "Quality and Style Guide" - we need **sample programs**, with rationale, for how to use Ada and how not to use Ada)
- Publicize the early availability of Ada 9X compilers - and make sure Ada 9X can co-exist comfortably with C/C++
- Develop and distribute training materials for all levels of personnel on software engineering and Ada
- Teach Ada as the primary language at the Military Academies and public universities (get the faculties to want Ada!)
- Deliver the benefits of Ada so the commercial sector will want to use it - make sure Ada really brings the promised portability, maintainability, reliability
- Get contractors to prefer Ada, and mean it when they bid Ada!
- Make Ada ubiquitous - free compilers, environments, interface software, libraries of software components

Participant Briefing
Jack Garman
NASA Johnson Space Center

(Presentation materials were submitted in handwritten form, a typed copy was created by the Ada Dual-Use Workshop staff.)

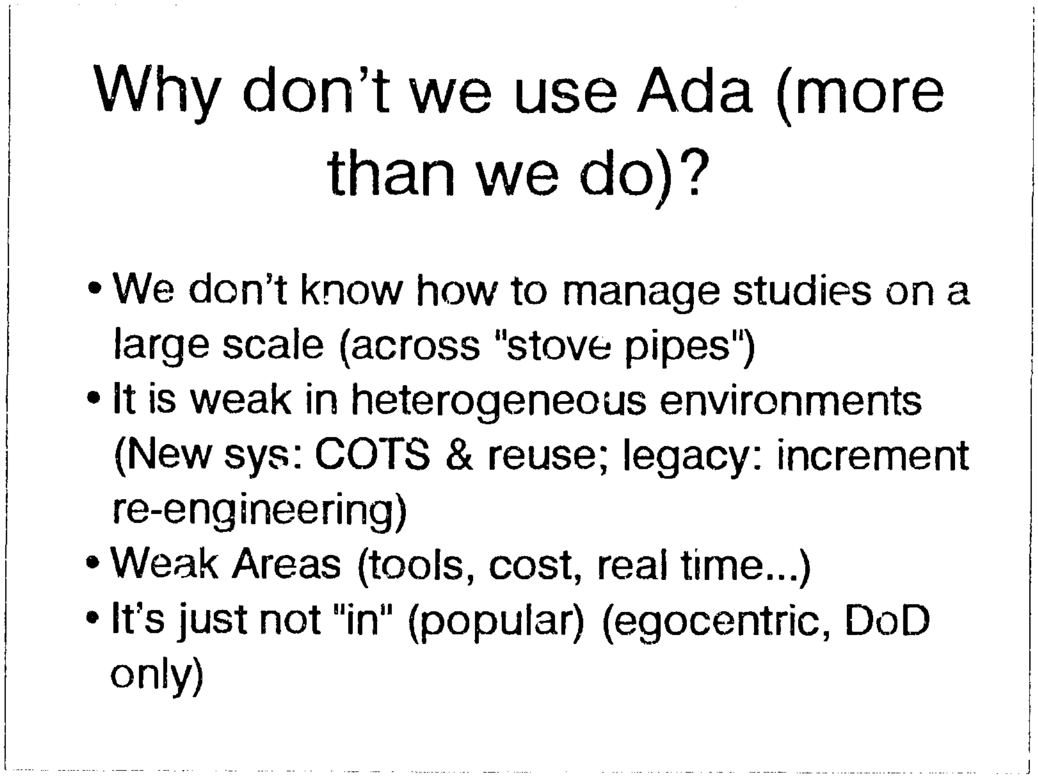
Ada Dual-Use Notes

Jack Garman
NASA Johnson Space Center

- Loosen up the rules (especially w.r.t upward compatible extensions)
- who wants to build something they can't "improve" (industry, other gov't, research, etc.)
- "if it works, don't fix it" but if it works its obsolete

- Subsidize (foster, support) libraries and buildings
- former is another/special form of reuse
- latter speaks to reality; heterogeneous systems and away from egocentricity

- Tell us (outsiders), unbiased, what Ada has done for DoD! (study, review)
- failures & issues as well as expected & unexpected benefits
- it might help DoD reassess, redirect, get its "house in order" w.r.t. Ada



Why don’t we use Ada (more than we do)?

- We don’t know how to manage studies on a large scale (across "stove pipes")
- It is weak in heterogeneous environments (New sys: COTS & reuse; legacy: increment re-engineering)
- Weak Areas (tools, cost, real time...)
- It’s just not "in" (popular) (egocentric, DoD only)

Participant Briefing
Richard Berg
Defense Mapping Agency

(Presentation materials were submitted in handwritten form, a typed copy was created by the Ada Dual-Use Workshop staff.)

Dick Berg
Defense Mapping Agency

CREDENTIALS:

I have 30 years software experience - none in Ada. I don't know what an Ada statement looks like (ends with a semicolon?).

I work in an agency of 7500 employees -- all of whom don't write software.

We make maps -- we don't write software.

We don't fund 6.1 and 6.2 research and development initiatives at all.

We wan software that works and is cost-effectively maintainable.

All our software needs are taken care of by contractors. Maintenance is provided by contractors, too. Actually we do have about 200 programmers who do a variety of small programming maintenance tasks, but not in Ada.

I don't really care whether Ada is the language of choice or not, but if I did, I'd have at least three reasons:

1. I believe you believe Ada is worthwhile, and I believe the business case studies.
2. I am responsible in DMA for ensuring DoD Ada policy is implemented.
3. It's the law.

With this background, I thought I could provide a different view of needs of Ada, and why I think it's important to improve Ada's appeal within the software community. Along the way I will comment on your three specific questions.

I want to talk in two areas:

1. Problems we are facing at DMA
2. Comments on what I've heard here so far and some recommendations (or perceived needs).

Description of DPS system.

- Ten year development
- Seven million LOC with 900 KLOC Ada
- Not critical life-dependent
- 60,000 embedded software errors
- Success but changing requirements

* Ada seems to be more cost effective to maintain, but we haven't made an effort to capture

significant metrics.

Real problem is to understand tools and techniques for software conversion/reengineering into Ada. (These could be 4GL Ada generators.)

Fund R&D initiatives in these areas if they don't exist.

Provide and advertise central, well-advertised location for information about Ada tools and techniques (Ada generally).

- Ada Information Clearinghouse?
- Internet "central site" repository?

Comments on three questions:

Many good comments from all presenters, but I really liked Judy Kernan's recommendation, and I support all of them. They were thoughtful and reasonable to implement, and represented a good summary of others' comments.

Regarding question 1 -- Making Ada the primary tool

- Perception problems here need correcting
    - Ada success stories
    - BYTE/MacUser/Govt Computer News...

- Real problems providing excuses for not using Ada must be solved
    - Incentivizing PMs to worry about life-cycle costs
    - Disincentivizing PMs to ignore Ada
    - Mr. Paige must stop permitting highly-placed DoD employees from disparaging Ada publicly
    - Clarifying DoDD 3405.1 regarding the "one-third" threshold

- Re-engineering needs must be addressed -- suspect there's a lot of code ripe for re-engineering into Ada
    - If there, need to publicize, make available
    - If not there, need to fund R&D to develop

Regarding question 2 -- Increasing commercial use of Ada

- I don't care about this
    - "Commercial use" means to me what vendors use to develop COTS packages I can buy
    - Not my worry -- market will determine what happens and we shouldn't try to influence it
    - If commercial product developers feel it's in their best interests to use Ada, they will. If not, they won't.

- I'll buy COTS packages in order to move to more "open systems" environment. Competition ought to keep the price down even if a producer uses C++ to develop and maintain a product.

Regarding question 3 -- Increasing Ada's appeal

- Change training and education paradigm
    - AT&T gave away C; now we're plagued with C programmers. Is there a model here? Low-cost ubiquitous development package is needed (Tiny-C, etc. off the MILNET)
    - Still teaching BASIC in elementary/secondary schools: work with SEEP (DDR&E) -- laws have changed which permit easier give-aways of DoD hardware and software

- Publish demonstrated business case studies.
    - Positives for Ada
    - Negatives for C and C++

Participant Briefing
Donn Milton
IIT Research Institute

(Presentation materials were submitted in handwritten form, a typed copy was created by the Ada Dual-Use Workshop staff.)

Donn Milton

I have been invited to this panel as a "government" participant, but actually my background spans the academic, vendor, and defense industry communities as well. So, in the interests of truth in advertising, I should let you know that I began my career as an academic, doing research in language and compiler theory, which I continued at Bell Laboratories. In late 1980, Ada was on the horizon, and I became Technical Director of CSC's Ada Integrated Environment Phase I effort for the Air Force. In 1982, I co-founded Verdix Corporation, served as Executive Vice-President, and managed the Ada Products Division. During that time I was also elected Treasurer of the ACM's Special Interest Group on Ada, which had grown to about 4,000 members. By 1987, Verdix had become profitable, but I became convinced that its continued success was essentially dependent on the strength of DoD's commitment to Ada, and that commitment was wavering. So I took a vacation--became a lawyer, and practiced bid protest law for several years. I also founded, and am now chairman of Pragma Systems Corp., which has been licensed by the Software Engineering Institute as a vendor of software process assessments. Today, however, I am addressing you as Director of the Tax Systems Modernization Institute, an FFRDC that was established earlier this year by IIT Research Institute to advise the Internal Revenue Service in the areas of strategic planning, acquisition, and technology assessment, related to the ongoing program to modernize the nation's tax systems. I should emphasize at the outset, that the views I express are my own, and not those of Verdix, SIGAda, Pragma Systems, IIT Research Institute, the IRS, nor the American Bar Association.

This is the first Ada-related conference that I have attended in about seven years, and I have to confess that I feel as if I have entered a time warp. The same people are here -- albeit a few years older and a few years more cynical -- but the conversations and the central issues have not changed one bit. The topics of today are precisely the topics of 1986: "when is the DoD *really* going to enforce the Ada mandate?" and, "what can we do to increase the use of Ada in the commercial sector?" It is both amazing and depressing to see how little real progress has been made in the Ada community, given the very real strides that have been made elsewhere in the computer industry.

I believe this lack of progress can be directly attributed to four factors. And I am going to address each of these because they provide us with valuable lessons learned, they identify mistakes that we cannot afford to repeat, and they serve to guide our investments for the future.

Firs and foremost, DoD has simply not enforced its own Ada mandate. The history of the Ada program is replete with waivers. The public perception is that by granting such waivers the DoD has demonstrated a lack of courage in its own convictions. Without such courage, DoD can have no rational expectation that Ada will be adopted commercially. I was particularly heartened yesterday to hear General Paige state emphatically -- and this is a quote -- "I am going to enforce it." Mandates are easy to write, they are only words. Enforcement is quite another matter. This commitment to enforcement that I heard yesterday, if effective, is a substantial and absolutely necessary step to the promotion of commercial use. I should add, however, that the enforcement must be pervasive. There should be a mandatory DoD FAR clause in every solicitation, in every contract, that requires all code to be developed in Ada. No exceptions should be made for Small Business Innovative Research programs, for ARPA programs, or for university research grants. Instead of bemoaning the fact that the research community has not embraced Ada, DoD can and should simply require that Ada be a prerequisite for all DoD-sponsored research. Moreover, DoD should further insist that every item of software it purchases -- whether source code or object code -- must be written in Ada. Imagine what would happen if DoD truly enforced the Ada mandate, and refused to buy any operating system, any word processor, any spreadsheet, unless it were written in Ada.

Second, DoD's investments in Ada technology have actively discouraged the vendor community. I remember back in 1983 or '84 when we were raising money for Verdix and developing our compiler, the Army announced that it was going to make the ALS compiler available for free. Needless to say, our investors expressed some concern. Fortunately, I was able to allay their concern with the observation that no software tool developed and distributed under Government contract had ever been much of a success. Unfortunately, it took the Army several more years, and untold tens of millions of dollars, before it abandoned the ALS as a failure. The other services made the same mistake -- the Air Force with the AIE, and the Navy with the ALS/N. In stark contrast, those vendors who developed their Ada compilers without the benefit of DoD contracts produced significantly better software and garnered most of the market, all at a small fraction of what was spent on the DoD-sponsored efforts. In technology, the proper role of Government is to establish standards, not to compete with private industry. DoD did an excellent job of establishing the Ada standard and the associated validation suite. It has done an abysmal job of managing the development of Ada tools. Imagine where we would be today if those untold tens of millions of dollars had been invested in both universities and defense contractors, earmarked for the acquisition - not for the development - of commercial-off-the-shelf Ada tools.

Third, DoD has failed to adequately assess and advertise the significant life-cycle cost savings that we all believe are obtainable with Ada. We have all heard anecdotal evidence of such cost savings. But even such limited evidence rarely makes it into the popular or trade press. Adequate funding must be provided to organizations such as the Ada Joint Program Office and Software Engineering Institute to collect in a systematic fashion data demonstrating the cost savings of Ada, and to perform and promote studies based on that data. While DoD must continue to enforce and reinforce the Ada mandate within the defense community, ultimately the commercial world will adopt Ada only after it is convinced that Ada will save money.

the foregoing factors have all addressed what DoD can do to rectify its pas mistakes. My fourth and final factor places some of the onus on industry. That is people resist change and therefore large numbers of short-sighted people who market Ada training have been able to convince the community that Ada cannot be used without weeks if not months of training in the proper use of the language and in the use of modern software engineering practices. I have heard some of these people despair that such training may likely not be effective for anyone over the age of 30. This conventional view that enormous up-front training costs are required for Ada is hogwash, but continues to intimidate potential DoD and non-DoD users.

From personal experience, I can attest that Ada is easier to learn than C, if only for the simple fact that Ada programs are much easier to read and debug than C programs. Anyone with programming experience in any third-generation language can obtain an adequate grounding in Ada in less than a week, self-taught, given a text book, access to a compiler, and proper motivations. Certainly there is a time and a place for training in Ada and software engineering, but the Ada training community must stop scaring away its customers before they are even in the door.

To conclude, I would like to address Parti t Question #1: "What actions can we take to make Ada the primary tool for developin vare systems within your government organization?" My government organizat the IRS, and this is not an easy question. The IRS currently maintains millions of lines o le that comprise the tax system. Until 1985, this was nearly all assembler code; in 1985, n w modules began to be written in COBOL. These millions of lines of code -- by and larg -- work. In contrast tot the software programs we see in the defense community, these mill ns of lines of code directly impact every taxpayer in this county. Also, each year the S devotes a considerable number of staff-hours to incrementally revise this code to accomm te changes in the tax laws and regulations. Now I firmly believe that if all of this code re to be entirely re-engineered in Ada, the risk-adjusted life-cycle savings would far igh the billions of dollars that such re-engineering would cost. But I can also g ee that Congress is not going to authorize these billions of dollars based on my pers assessment of risks and life-cycle costs. For Ada to be adopted by organizations such as the IRS, organizations that are saddled with substantial legacy software systems, you must come up with proof beyond a reasonable level of doubt that re-engineering in Ada will be cost-effective.

Participant Briefing
1Lt George Tadda, US Air Force
US Air Force

(Presentation materials were submitted in handwritten form, a typed copy was created by the Ada Dual-Use Workshop staff.)

Ada Dual-Use Workshop

George Tadda
SSC, Gunter AFB
Montgomery, AL

Recommendations

- Concur with most of the findings presented
- Define and Publicize DoD's goals and objectives that require Ada

Recommendations (Cont)

- Encourage use of Ada in non-DoD systems by:
  - Advertise Ada Successes
  - Support companies using Ada
  - Support DoD systems use commercially
  - Educate companies in the advantages of Ada

Comments

- Provide levels of validation for Ada Compilers
- AdaSAGE for DOS

Position Paper

Eileen Quann
FasTrak Training, Incorporated

Eileen Quann

Recommendations for Expanding the use of Ada within DoD

1. Either enforce the Ada mandate or eliminate it. Don't assume the absence of waiver requests means that DoD is using it. It may only mean that few people are requesting waivers. Many people ignore the mandate or assume it doesn't apply to them. Specifically, since you don't have the staff to check out every line of code being developed, I recommend the following approaches:

   a. Identify every program under development at DISA. Perform a random audit of the software being developed. Ask to see the code. Do the same thing at DLA and the other defense agencies. Do the groups developing software in other languages have waivers or does the mandate not apply to them? Do something about your discoveries.

   b. Require that requests for training in C or C++ only be approved for programs that have approved waivers, unless the mandate doesn't apply to them. Check 1556s for compliance. Check 1556s for the last six months.

   c. Read the CBD. Perform word searches on Ada, Software Engineering, Object-Oriented, C++, compilers. Identify DoD procurements that are requesting compilers for C++. Identify DoD software development efforts that indicate a language other than Ada. Do they have an approved waiver? Why is DoD issuing requests for tools, compilers and development in other languages?

2. Clarify the Ada mandate. Who, explicitly, does it apply to? What organizations/applications are not included in the mandate? The number of DoD people who claim that "the mandate doesn't apply to us", is astonishing.

3. Broadcast Ada success stories. Promote technical seminars (perhaps about two hours in length) on a regular basis (quarterly?) that identify programs being implemented in Ada within an agency. Advertise the programs agency-wide. Encourage attendance. Show the metrics. Teach the lessons learned. When warranted, give the project (or individuals) an award or bonus. (for reuse, high productivity, high reliability, quick response?) Ada needs more visibility than just being mandated. Have General Paige or General Short sponsor the seminars and attend them (in DISA for example). Show that you really are committed to Ada.

These recommendations are specific, inexpensive and easy to implement. Each one will show commitment, something that may be perceived as lacking in DoD. I will be happy to discuss any of these ideas further. I can be reached at (301)924-0050.

Position Paper

William Thomas
Adirondack Computational Labs

Adirondack Computational Laboratories Inc.

100 Crossways Park Drive West, Suite 112
Woodbury, N.Y. 11797

Tel: (516) 496–0876
Fax: (516) 496–0911

The Ada Technology Incubator

The Ada community has benefitted enormously from the work of the Ada Software Alliance (ASA). It looks like the ASA's recent Ada marketing campaign is poised to blaze new trails into the marketing frontier. This campaign coupled with the general medias increasing curiosity with Ada9X should prove to be a market increasing situation. Indeed the ASA train is heading down the right track and it seems that the political and marketing end of this technology is definitely being cared for.

The Ada Technology Incubator (ATI) is (or could be) an entity that focuses on increasing the Ada market and creating new Ada business through the direct injection of Ada technology into the commercial sector.

The ATI could ::

- Act as a showcase for all Ada technology, without marketing pressures, on a level playing field.
- Provide a channel for Ada vendors to offer tools at a reduced cost to qualifying commercial efforts.
- Aid commercial efforts in porting products to a wider variety of platforms.
- Nurture and support new startup companies selected for their potential to increase the Ada market.
- Give birth to a newsstand quality Ada magazine.
- Act as the needed seedbed and marketing medium for the software components industry.

The ATI should be ::

- Stocked and highly supported by all Ada vendors.
- Funded by the government.

Act as a showcase for all Ada technology, without marketing pressures, on a level playing field.
Perspective commercial users could benefit from a place that showcased a large selection of Ada technologies thus allowing a user to taste and test and without the usual sales pressures. Vendors would benefit from "displaying" their wears in a hands on environment. Potential users might be able to receive a few days of free Ada training as a result of several Ada training vendors offering one day "samples" of their curriculum. Component manufactures could benefit from an environment that can demonstrate first hand portability of their products across a large number of platforms.

Provide a channel for Ada vendors to offer tools at a reduced cost to qualifying commercial efforts.
The initial costs of adopting Ada technology can be quit high, especially in a startup situation. All would benefit from putting low cost Ada technology into the hands of those would need it. Keep in mind that it is a very risky thing for an Ada vendor to venture out and drastically reduce product price in order to increase market share. If not successful the resulting decreased profit margin from the existing customer base could prove disastrous. What's needed is a channel for vendors to offer their products at a reduced cost (without jeopardizing existing profit margins) to customers who could otherwise could not afford to use Ada in their business ventures.

Aid commercial efforts in porting products to a wider variety of platforms.
If a commercial company has the good sense to develop their product in Ada then the Ada community should feel obligated and glad to help ease the cost and effort needed to get that product running on as many platforms as possible. For the vendors lucky enough to host the application it's a feather in their cap, if the application is successful chances are the developer will make the full investment needed to support the product. The ATI could serve as a neutral porting ground for such efforts.

Nurture and support new startup companies selected for their potential to increase the Ada market.
If a group of bright individuals pooled together to develop a set of software components, what organizations are in place to directly help them get on their feet. The mere existence of an ATI could help tremendously in attracting potential investors. To the eyes of an investor an entity like the ATI helps to minimize the risk of the venture, it helps validate the concept.

Give birth to a newsstand quality Ada magazine.
I see an entity like the ATI been used to spawn a news stand quality, highly circulated Ada magazine that would compete with the likes of *C++ Report* and *Journal of Object Oriented Programming*, a magazine that would reflect the professionalism found in the Ada culture. The magazine could serve as an inexpensive marketing medium available to supporting vendors, as well as a medium for any Ada vendor. The magazine would help validate the Ada market and provide a sustained presence of Ada within the larger mainstream software marketplace. The magazine could also serve as a sorely needed marketing medium for the software components industry, software components could be advertised in the same medium as their supporting compilers.

Act as the needed seedbed and marketing medium for the software components industry.
Developing components to run on as many platforms as possible is the ideal situation for the developer, for the tool vendors it pays to have as many components available as possible that add value to their products. The ATI could serve both component developers and compiler vendors by providing a rich environment where components can be developed, tested and validated across a large number of platforms, a situation that would otherwise be impossible to achieve by the developer alone. The component developers could then use the previously mentioned Ada magazine as one of their marketing outlets. Tool vendors could guide the component developers by suggesting specific API's and other components needed by their customers and that add value to their products.

The ATI could be a shinning example of how vendors, although direct competitors, can join forces to increase market share and promote their common technology.

Ada in Academia

To any willing university, college, technical school, or high school, offer the following::

The Ada curriculum that ARPA is working on,
The Ada technology of their choice,
The hardware if it's needed,
And what ever training is needed to teach the teachers.

Stipulate that they must implement the curriculum, and at some time in the future must participate in a nationwide test geared to determine the effectiveness of the effort. The test should take the form of a "National Software Engineering Competition", it should be team oriented, and offer prize money in amounts that can not be ignored.

I've dealt with high schools concerning the subject of Ada and belive me the carrot has to be wrapped in a wad of bills. When it comes to furthering Ada in academia you have to make them an offer they can't afford to refuse.

Code Quest

Develop a (possibly multimedia) educational "adventure game" (hopefully in Ada) and call it say "Code Quest". The object of the game could be to accumulate knowledge and correctly answer a large number of questions pertaining to Ada9X. The user would reach plateaus one of which would yield a 'token' redeemable for an Ada compiler of their choice. You could make it so that reaching further plateaus would require the use of an Ada compiler and hands on programming experience. You should be able to produce enough material to make the courseware require at least 40 to 80 hours of interaction before yielding its ultimate prize of say a 1000 to 5000 dollar scholarship for the first say 1000, to complete the quest.

Position Paper

Dick Dye
CTA, Incorporated

CTA
INCORPORATED

Subj: Inputs to the Ada Dual Use Workshop
To: Ada Dual Use Workshop
CC: Merle Danskin
John Marciniak
Don Reifer
From: Dick Dye 719-590-5224
Date: October 13, 1993

I am unable to attend the Ada Dual Use Workshop. These are the inputs I would have conveyed in person to the Industry Panel.

Personal background relevant to the Workshop:

- I have been working with Ada since 1980 when I was a member of the United States Air Force Academy's first Ada research team.
- I have been working with Ada full time since 1987. In the last six years I have:
  - Designed and built Ada systems for real-time and non-real-time applications.
  - Consulted on numerous Ada projects.
  - Worked on the Ada/C++ Business Case Analysis in 1991.
  - Taught Ada classes as a TRW and CTA employee and as a part-time instructor for two different colleges.
- I have been very active in the Association for Computing Machinery's Special Interest Group for Ada (ACM SIGAda).
  - I have been the organization's secretary for two years.
  - I have twice been the Tutorial Chair for TRI-Ada, the major conference sponsored by SIGAda.
  - I am the Program Chair for the next TRI-Ada.
- Summary: I am well informed of multiple aspects of the Ada world:
  - The technical side of the language and how to apply it to large software systems.
  - The teaching of the language and the issues around it.
  - The politics and marketing of the language.

1. What actions can we take to get Ada accepted as the preferred language for development of software within your organization?

- My company and all the companies I have worked with respond to the customer's direction. DoD contractors will do whatever the DoD asks them to do. If the DoD says "do it in Ada," we will. If the DoD asks "what language do you recommend?" we will probably recommend C or C++ (for reasons that are not essential to this discussion). This does not mean it is in DoD's best interest for us to use C or C++. The Business Case Analysis determined that development costs are roughly equal (within 10-15%) for Ada and C++. The data is not available yet for life cycle costs. I believe that Ada will be proven to be more cost-effective for maintenance than C or C++.

Recommendations:

  - Enforce the Ada mandate. Contractors will respond to government direction.
  - Start to collect the data necessary to do a life cycle cost analysis of Ada and other languages.
  - Don't separate development contracts from maintenance contracts. The key to winning a contract is lowest credible cost. By including the maintenance cost with the development cost, I believe Ada will prove more cost-effective.

- The standard accounting practices established by the Financial Accounting Standards Board (FASB) require that the cost of developing software be expensed in the year the cost is accrued even though the software will be used as a capital asset in the future. If the rule were changed so that the cost of developing software could be depreciated over the life of the product, it would be more economical to develop software.

  Recommendation:

  - Work with the FASB to change Financial Statement 86 so that software developed in a rigidly controlled, internationally standardized language can be depreciated rather than expensed.

2. What investments can we make to increase the commercial use of Ada?

   - The two biggest arguments I hear against Ada are that the existing software base is not Ada and that it is hard to interface Ada to non-Ada products.

     Recommendations:

     - Using the New York University Gnu Ada Translator (GNAT) project as a model, fund efforts to develop the bindings from Ada to the most commonly used products.
     - Continue to fund the development of the Ada 9X Appendix M, "Interface to Other Languages."

3. What can we do to increase Ada's appeal, adoption, and popularity within your software community?

   - The appeal of Ada is limited to developers with the wisdom to appreciate the discipline that Ada enforces on their software. Not many developers appreciate that discipline. Therefore, the issue today is not to improve Ada's appeal or popularity but to improve Ada's adoption. Given the desires and sophistication of most developers I meet, adoption is promoted from above.

     Recommendations:

     - Emphasize the long term quality and life cycle cost of software products in the initial procurement cycle by procuring both the development and maintenance phases at the same time.
     - Enforce the Ada mandate. The fact that no Ada waivers are granted does not mean that Ada is being used exclusively, it means that the waiver is not requested and the lack of a waiver is not enforced.
     - Continue to fund the Ada Software Alliance's marketing effort into the commercial sector emphasizing the life cycle savings and higher quality software possible with Ada.
     - Collect and promulgate the metrics that prove Ada's long term benefits in lowering life cycle costs.

Summary of recommendations (in order of importance):

- Enforce the Ada mandate.
- Collect and promulgate the metrics that prove Ada's long term benefits in lowering life cycle costs.
- Don't separate development contracts from maintenance contracts.
- Work with the FASB to change Financial Statement 86 so that software developed in a rigidly controlled, internationally standardized language can be depreciated rather than expensed.
- Fund efforts to develop the bindings from Ada to the most commonly used products.
- Continue to fund the development of the Ada 9X Appendix M, "Interface to Other Languages."
- Continue to fund the Ada Software Alliance's marketing effort into the commercial sector emphasizing the life cycle savings and higher quality software possible with Ada.

Position Paper
Jeffrey Hickey
Advanced Software Technology, Incorporated

Prof. Michael Feldman
Dept. of Electrical Engineering and Computer Science
George Washington University
Washington, DC 20052

Dear Michael:

Thank you for inviting me to comment on the state of Ada today. I looked at the four questions you sent, and thought about how to answer them. I realized that my answer to each of them was the same:

"Increase the viability of Ada as a commercial programming language"

As an adjunct professor in computer science for some local colleges and universities, I am not in a position to determine curriculum. Therefore, I am asked to teach courses that these institutions feel will attract students (read "make money"). This is especially true in the continuing education courses that I normally teach. Since these courses are attended by people who want to improve their skills (usually with an eye to improving employment status), they are definitely market driven. After putting Ada courses in the catalogs in past semesters, it has become obvious to me (and the colleges!) that the general programming public has no interest in Ada training. As a result, I have been informed by one university that these courses will no longer be offered if there is no attendance in the spring semester.

We can say all we want about the usefulness of Ada as a teaching language (it's great) and the uselessness of C/C++ (they suck), but unless people can get jobs programming in Ada, it will remain a curiosity, not a viable programming language. The colleges cannot get enough people to teach the C/C++ courses; they are always full. A quick perusal of the Sunday want ads will show far more jobs available in C/C++ than Ada; in fact, Ada is hardly mentioned. If you were a student trying to get a job in this tough market, which language would you choose?

Given the above, what can be done about getting Ada into the programming mainstream? Step number one is that the DoD has to stop their two-faced treatment of Ada. If the language is mandated, then it should be used almost exclusively. This is certainly not the case. In many projects, it is a given that Ada will NOT be used; the mandate is totally disregarded. If the "champion" of the language has such mixed feelings about it, how can we expect anyone else to wholeheartedly adopt it? I personally feel that this has a lot to do with the current commercial market. If I was in the military writing programs and expected to get a civilian job when I left, which language would I want to be learning? (Hint: it isn't Ada).

Second, we need more publicity about Ada. The Metcalfe article in InfoWorld was great; we need more like it. We need companies who have had commercial success with Ada to be more public about it. We need to be more visible at conferences and trade shows. Again, COMDEX was a great start, what are we doing for an encore? Should this be a DoD job? Yes, if they are really committed to Ada. This will attract both new and experienced talent into the Ada world, which can only benefit future DoD projects in Ada. In addition, the commercial Ada companies must realize that their future is in the commercial marketplace. Therefore, they must learn to market themselves (read: spend money) to sell compilers, instead of depending on a mandate to do it for them. Toolset development is essential also; programmers in other languages laugh at the primitiveness of most Ada tools. Let's stop be being language lawyers and develop some useful tools instead.

I view Ada 9X as the last great hope of getting Ada into general usage. All of the above suggestions are not new or novel; most are being done. However, the scope of the effort is too small. Ada 9X is a viable alternative to C++, but if we don't let people know, it will fall by the wayside.

I would suspect that this is not quite what you wanted, Mike, and I apologize if I sound too much like Greg A.. I f you do not think these comments are appropriate, I will not be upset if you don't pass them on. I get a little emotional about Ada; I don't like it when good solutions are ignored for the wrong reasons.

Sincerely,

Jeffrey Hickey
President
Advanced Software Technology, Inc.
5 Patricia Lane
Patchogue, NY 11772
(516) 758-6545
jeffhick@src4src.linet.org

Participant Briefing
Audrey Hook
Institute for Defense Analyses

POSITION STATEMENT

ORGANIZATION TYPE: FRDC

The primary mission of the Institute for Defense Analyses(IDA) is to assist the Department of Defense address important National security issues of both immediate and enduring concern. In carrying out this mission, IDA has been involved in assessing technology that enables the department to maintain military superiority. Support for the Ada program has been one of these enabling technologies. Now it is becoming clear that economic considerations are of importance for National security as we are beginning to be involved in the development of National enabling technology statements. Our community of users are (1) in the R&D business (2) policy or program analysis business (3) managers of large military or business projects. I have paraphrased the questions asked of government panel participants to fit our organizational experience with these three communities of DoD clients.

Question 1: *To what degree is Ada encouraged/required by our DoD clients or by the in-house development staff?*

Our clients who are in the R&D business or policy/program analysis business have two immediate concerns, development cost and time-to-deliver usable software. The programming language used is of very little concern except that less programming is better than more. What is generally wanted is a "prototype" where performance is of less importance than demonstrations of concepts, requirements, and simple tools for the user. The solution for these clients is usually COTS with minimal programming in C or Visual Basic. A client who is managing a large military or business project may or may not be interested in "prototyping" but, if prototypes are done they must be done quickly at minimal cost.

The frequently heard comment from our clients and our in-house staff is:

"When we can buy a Turbo Ada compiler for $300, libraries of functions and bindings for another $100, then we will seriously consider using Ada for prototyping."

Clients who do not prototype but are using Ada because of DoD policy are now being approached by contractors who are riding the Object Oriented bandwagon. Project managers are somewhat confused about OOD or OOP but they have been told that Ada is incompatible with both.

Recommendation:

(1) Invest in the GNU Ada compiler (execution version) and make it available to anyone who will pay a nominal fee for its distribution.

(2) Analyze the operation of a successful C/C++ company (e.g. Turbo). Look at what they are providing (libraries), how its being priced and distributed, how it is being advertized, and product improvement cycles. Copy what they are doing to produce "Turbo" Ada.

(3) Focus software engineering education on project managers who need to be conversant with Ada facts and latest fads in methodologies.

Question 2: ***What can we do to make Ada the primary tool for developing software systems?***

The plain answer to this question is that there is no "silver bullet" but there are some policy practices that can help level the playing field. There has been a perception that because Ada is a DoD language it is the only one that has a compiler validation requirement. The fact is that all FIPS languages have compiler validation requirements and all software that enters the Federal Government inventory must have been produced by a FIPS language compiler that has *current* validation status. This requirement is not being enforced or else we would not have software inventories for obsolescent versions of COBOL and FORTRAN, C and C++.

Recommendation:

(1) Revise DoDD 3405.1 to include the requirement for maintaining validated status for all FIPS compilers, or,

(2) Issue this policy in some other appropriate form.

Question 3: *What can we (DoD) do to increase Ada's appeal, etc. for dual use?*

Some small things have been done to decrease the perception that Ada is a language that is designed for use only within Defense. One of these actions is the recent MOU with the Department of Commerce (NIST) by which Ada validation certificates carry both DoD and NIST logos. Another action was the declaration by the AJPO that the Ada Compiler Validation Capability (ACVC) is in the public domain and exempted from export controls. However, "advanced software environments" using Ada examples are still on the Miliary Critical Technology List (MCTL). The whole policy of technology control is undergoing change in light of world-wide developments. It is an advantageous time to ensure that Ada and Ada tools or re-usable software are considered as National enabling technologies rather than technology to be held close by only DoD and DoD contractors.

Position Paper

Kenneth Russell, Howard Stewart, and Paul Whittington
EG & G IDAHO

Proposal
The Future Success of Ada 94

19 October 1993

Although Ada 83 is an excellent language for a variety of application domains, it has not enjoyed commercial acceptance. Ada 94 offers system developers a current technology language with facilities such as object orientation which are in high demand. By considering the lessons learned from the history of Ada 83, Ada 94 may be afforded greater acceptance by the government and the commercial sector.

After the initial excitement over the introduction of Ada 83, two important barriers to its acceptance as a full system development language became obvious. These were:

1. The Ada compilers were not mature and contained errors and inefficiencies which were not present in commercial quality compilers for other languages.

2. There were rarely any packages or tools available in the domains required by full system developers. It was after all, "just a programming language", not a development environment.

Ada 94 is well designed offering modern structures and methods. It has been favorably compared against the C++ language as a desirable language for large system development using object oriented programming techniques. With proper ground work, Ada 94 could be accepted as a viable full system development language upon its release, and the experiences with its predecessor, Ada 83, would not be wasted.

1. The current compiler vendors have years of experience in Ada and are improving their products. With this experience, and the paths paved by the GNAT compiler, the technology to produce commercially competitive compilers is available. If these compilers also offered a quality development environment and tools, they would be commercially competitive, and the resulting economic competition would provide incentives for further improvements.

2. Ada 94 should be released with a rich set of reusable components (objects and classes) which provide the developer with a useful development platform. With the incorporation of objected oriented methods, these tools should be designed to be further enhanced and complemented by the original developers as well as any commercial developer wishing to provide third party tools. Since AdaSAGE is one of the most widely accepted and successfully used development tools in Ada, and it is sponsored by the U.S. government, it could be enhanced to provide object oriented capabilities for Ada 94 which would promote immediate acceptance by the many programmers now experienced in its use. It could be made available at no cost to government and commercial developers to provide a rich set of tools and a basis for further development.

Object Oriented AdaSAGE Development

14 October 1993
Idaho National Engineering Laboratory

Howard D. Stewart
Paul H. Whittington
(208)526-9103

AdaSAGE is currently composed of abstractions implemented as modular packages and stand alone processors which provide a development tool set for application programmers. The capabilities of Ada 94 will be utilized to recast the existing abstractions into classes. The new object oriented AdaSAGE would take advantage of reuse through inheritance and polymorphism. Associated tools would also be revised to present an extensible object oriented development paradigm. Insure compatibility with available standard contract 9X (94) compilers.

Time of development	=>	12 months
Cost of development	=>	$825k

AdaSAGE™ Notes

Howard D. Stewart
Kenneth D. Russell
Paul H. Whittington
Idaho National Engineering Laboratory
Special Applications Unit
P.O. Box 1625, Idaho Falls ID 83415-1609

September 16, 1993

Abstract

AdaSAGE is an application development tool for Ada programmers implemented as a set of Ada packages and a set of executable programs used as support utilities during application development and operation. Many questions have been asked regarding the characteristics of AdaSAGE (i.e. Is AdaSAGE relational?). Herein several such questions are posed and answered.

What is the purpose of AdaSAGE?

AdaSAGE was designed to provide tools and an environment for Ada programmers to develop major non-proprietary systems completely in Ada which are as good as or better than systems developed using alternate methods.

Who uses AdaSAGE?

The SAGE system began at the INEL (Idaho National Engineering Laboratory) in 1982. In 1987, it was made available in the Ada programming language for use on U.S. Marine Corps projects. Since then, it has been used by all of the major services of the DOD (Department of Defense), the DOE (Department of Energy) and private industry. It is available through a variety of distribution channels to both government, academia and private industry.

Where can one get AdaSAGE?

At the present time, AdaSAGE may be acquired from one of the following sources:

AdaSAGE Users' Group
Idaho National Engineering Laboratory
(208)526-0656

RAPID Center
Sgt. Stephen St. Espirit
(703)285-9007

Office Of Scientific Technology Information
P.O. Box 62
Oak Ridge, TN 37831
(615) 576-1166

Is AdaSAGE written in Ada?

AdaSAGE reusable Ada packages are 100% Ada code. In the MS-DOS environment, approximately 5% of AdaSAGE is written in assembly language to provide optimal interface to low level hardware that is not readily available to MS-DOS.

How extensive is AdaSAGE?

During AdaSAGE development, extensive effort is expended to create application development tools and associated system development methodologies that maximizes the effectiveness and productivity of Ada system developers.

The AdaSAGE system is comprised of over 300,000 lines of code representing over 250 person-years of effort with an average cyclomatic complexity of 4, as measured by a standard code metric analysis tool.

Does AdaSAGE support the Open System Environment?

The AdaSAGE development team is committed to supporting the Open System Environment standard. At the current time AdaSAGE provides support in the following ways:

Ada	AdaSage is written in Ada.
SQL	Interactive and imbedded SQL is provided.
GKS	Standard GKS is provided.
POSIX	AdaSAGE is available on POSIX systems.
GOSIP	AdaSAGE supports multi-user development on GOSIP networks

On what platforms is AdaSAGE available?

Versions of AdaSAGE tools and reusable packages are available within a variety of environments. AdaSAGE is now available or will be available soon on the following hardware platforms:

Intel 8088
Intel 8086
Intel 80286 Real and Protected Mode
Intel 80386 Real and Protected Mode
Intel 80486 Real and Protected Mode
AT&T 3B2
Sun SPARC
RS/6000

AdaSAGE runs under the following operating systems:

MS-DOS
UNIX
AIX

The AdaSAGE application development system is available for the following Ada compilers:

Alsys
Meridian
Verdix

Can AdaSAGE be used to develop large systems?

Since AdaSAGE is written in Ada, it supports the development of very large systems. In practice, it has been used for the development of small to very large systems, supporting both large program and capacity requirements. Some of the capacity constraints follow:

Category	Constraint
Relations (flat files) per data dictionary	1,000
Attributes per relation	500
Indexed attributes per relation	500
Forms per data dictionary	6,550
Record (tuple) size (bytes)	32,000
Field (attribute) size (bytes)	
Fixed length	32,000
Variable length	2,147,483,000
Significant digits per number	60
Records (tuples) per relation	2,147,483,000/RecordSize
Name length (characters)	312

Are training and support available?

Beginning, advanced and graphics training courses are available at the INEL or at a requested location if a sufficient number of students is available. The AdaSAGE Users' Group supports registered users with a technical support line, new release upgrades, a bulletin board, a newsletter and other training and assistance. For more information, contact the AdaSAGE Users' Group at (208)526-0656.

Is AdaSAGE maintained?

AdaSAGE is supported by the Idaho National Engineering Laboratory (a DOE laboratory) with funds supplied by the AJPO (Ada Joint Program Office) and other DOD agencies. A long term agreement is in place between the DOE and DOD for continual support in the future. Work is currently underway for conversion to other computer platforms, open system architecture adherence, and other areas.

Does AdaSAGE have a Data Dictionary?

A data dictionary contains the data structure (schema) and often other ancillary information about the data and its structure. AdaSAGE supports an independent data dictionary and a facility for defining, modifying, and reporting on this information. This capability allows many different applications to access the same data dictionary and provides physical and logical data independence as specified in rules 8 and 9 of Codd's Rules.

What security support is included in AdaSAGE?

AdaSAGE provides many levels of security. Even though data integrity checks are a form of security and provided by AdaSAGE, they will not be discussed here. Access security is provided by password protection at the database, relation (flat file), and the individual attribute (column) level. These are defined within the data dictionary and enforced throughout the system for both programmers and users. Additional security is often required within applications. Many such applications have been provided using AdaSAGE with additional protection through user tables which contain user identification information, statistical information, and network and system information concerning user logon identifiers and privileges.

Does AdaSAGE support the development of relational data bases, and if so does it support Codd's rules?

One of the many features of AdaSAGE is its support for relational data base development. E.F. Codd, a mathematician, first defined the relational rules which were presented in a paper published in ACM Communications, June 1970. Later, in 1985, he summarized a foundation principle and twelve rules which should be adhered to by a relational system. He stated that a database system fulfilling at least half of these rules and having the capabilities eventually to fulfill all of them can be considered relational. The following table indicates the adherence to these rules by AdaSAGE, ANSI SQL (the source for FIPS 127 and FIPS 127-1), and IBM's DB2 which is considered by Mr. C. Date of the Relational Institute to be the leading SQL representative data base.

Codd's Rule	Description	AdaSAGE	ANSI SQL	IBM DB2
0	Foundation Principle	P	P	P
1	Information Rule	Y	Y	Y
2	Guaranteed Access Rule	Y	N	P
3	Missing Information Rule	P	N	P
4	System Catalog Rule	P	N	Y
5	Comprehensive Language Rule	P	P	P
6	View Updatability Rule	N	N	P
7	Set Level Updates Rule	P	P	Y
8	Physical Data Independence Rule	Y	Y	Y
9	Logical Data Independence Rule	Y	P	P
10	Integrity Independence Rule	P	P	P
11	Distribution Independence Rule	P	?	IS
12	Nonsubversion Rule	Y	?	Y

Y = Yes N = No P = Partial IS = Intended Support ? = Unspecified

Does AdaSAGE speak SQL?

Speaking SQL usually refers to an RDBMS. AdaSAGE is not an RDBMS. AdaSAGE is an application development tool that provides facilities for creating an application specific relational data base.

There are two aspects of SQL dialog to consider. First is listening to SQL and responding by executing the requested command. Second is issuing SQL to get a foreign system to execute some process on your behalf. In the first case AdaSAGE provides both an imbedded SQL technology and an interactive SQL system adapted to comply with ANSI-SQL DML Level 1. In the second case AdaSAGE does not provide any capabilities for creating SQL commands, but since AdaSAGE is a set of Ada packages there is no reason that a package could not be developed to do so.

Does AdaSAGE support multiuser system development?

AdaSAGE provides for both single user and multiuser systems. Both types of systems have been represented in fielded applications. A unique feature of the AdaSAGE multiuser system (MultSage) is the distribution of not only the data to be shared by many users, but also the distributed processing concept where each user contributes his CPU to the multiuser database management and locking functions. This concept is used for LAN (local area network) type applications.

Are there audit trail and recoverability features in AdaSAGE?

The capability to record all transactions and roll forward from previous dates gives an audit trail and recover capability. These features are often provided within data base management systems, and are provided with AdaSAGE as a logging option, but seldom if ever are they used in final applications because of the excessive time and data storage requirements. An application using AdaSAGE may be developed to provide full or partial capabilities in this area using the Ada language and AdaSAGE provided packages. More often, applications regularly backup or copy current data to another device to provide data recoverability. As well, AdaSAGE provides a facility to rebuild or recover current data base information in place.

Will AdaSAGE interface to DBMS's?

Neither Codd's rules for a relational system nor ANSI SQL specifies the physical storage format of data. Logically, it must appear to be two dimensional and is often referred to as being composed of relations (or a less precise term, flat files). Each data base system, therefore, tends to have its own unique method of physical data storage. A method to move data between different systems is often required. There are some commonly used formats of files from which many DBMS's load and unload data. This is one method of interface between two DBMS's and is supported by AdaSAGE. With AdaSAGE, the Ada language may also be used to directly and much more quickly move data from a DBMS into an AdaSAGE relation or in the reverse direction. The only requirement is that the DBMS support an Ada interface.

Is AdaSAGE a code generator?

Currently there are no components of the AdaSAGE system that generate code. The application paradigm supported by AdaSAGE results in applications that are data driven. This data is contained in a data dictionary containing information about the relational structure of the application's data, the application's user presentation and various other application specific information.

Are AdaSAGE packages reusable?

AdaSAGE provides many Ada packages, all of which are reusable. Many of them are independent and may be used within any Ada system without causing the inclusion of the data base or user interface facilities. These include libraries for sorting, data movement, binary operations, bit manipulations, graphics, string manipulation, and others. There are also packages which directly access the data base system and others which access the user interface facilities.

AdaSAGE adheres to the concept of "black box" modularity which emphasizes the reuse of code through data encapsulation and extensibility through inheritance.

All of the provided Ada packages are designed to be understandable, useful and usable, and are loosely coupled and highly cohesive at their level of abstraction.

Does AdaSAGE support graphics development?

AdaSAGE graphics support is available on some platforms. The graphics support includes the following:

Low level support
ANSI-GKS Level ma

Intermediate level support
Primitive Calls - Rotatable Arc, PolyLine, Rectangle, Rotatable, Scalable Text and Bit-Mapped Text
Input Calls - Keyboard and Mouse
System Calls
Window Calls
Device Calls
Font Calls
Locator Calls
PCX Graphics File Calls

High level support
2-D Graphing Package - Pie, Bar, X-Y, Area and Line Charts
Graphical User Interface Package - Menu bar, Pull-down Menus and Pop-up Menus
Data driven graphical reports edited in THOR

Device support
Video - CGA, EGA, VGA
Printers - EPSON-FX, EPSON-LQ, HPCL, Postscript
Mouse - Microsoft Mouse Driver

Support Utilities
Resource Compiler
Stroke Font Editor
Bit-Mapped Font Editor
Cursor Editor

This material resulted form work developed under Government Contract No. DE-AC07-76ID01570 at the Idaho National Engineering Laboratory and is subject to "Limited Government License". Neither the United States nor the United States Department of Energy, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately-owned rights.

APPENDIX E

ADA DUAL-USE WORKSHOP
OCTOBER 19-20, 1993

INDUSTRY PANEL PROCEEDINGS

November 8, 1993

Defense Information Systems Agency
Joint Interoperability and Engineering Organization
Center for Information Management
Software Systems Engineering Directorate

TABLE OF CONTENTS

1.0 INTRODUCTION

This appendix contains the findings, conclusions, and recommendations for the three questions that were posed to the Industry Panel. This data supports the summary of the industry panel, as discussed in the Ada Dual-Use Workshop Summary.

2.0 QUESTION #1: What actions can we make to get Ada accepted as the preferred language for development of software within your organization?

FINDINGS (We Heard)

The following findings were the result of participant briefings:

- The mandate is weak, not consistently applied, and often ignored by our customers.
- More case studies and global statistics which substantiate the business case for Ada are needed.
- More bindings (especially to standards like Windows) and COTS tools are needed.
- Better education and training for managers and programmers (both Government and Industry) is needed.
- Better marketing and communications are needed to promote and sell Ada.
- Availability of Ada 9X tools, training, and implementation guidance needs to be accelerated to meet the window-of-opportunity within industry.
- Incentives are needed both to defer cost and increase profit for those who demonstrate Ada excellence.
- The means to make more effective use of reuse repositories is needed.
- Educational use needs to be stimulated by permitting language subsets and supersets.
- Pilot projects to demonstrate proof of concept of new ideas should be performed.
- Ada should be incorporated into SEI Capability Maturity Model's Software Process Assessments and Software Capability Evaluations.
- The industry should be stimulated to develop and recognize core competency in Ada.

CONCLUSIONS (We Think)

Based on the session findings, the following conclusions were made by the panelists:

- Testimony supports the fact that Ada has been successful in mission-critical, software-intensive systems within DoD and elsewhere. We should not corrupt this support in the name of winning a popularity contest or by moving Ada into a market it was not designed to support.
- We need to shore up the perception of an eroding base for Ada within the DoD. Solicitations in mission critical systems which do not call for Ada should not be allowed.
- Our strategy should look towards coexistence rather than exclusive use. A coexistence approach with other languages such as C and C++ based products (COTS and reuse) in their technically appropriate area of application should be evaluated.
- The "software engineering market" is very different from the "general software market". We need to focus on the market that Ada was designed for (the mission-critical systems area) and not be diverted to entering a highly competitive and price-driven marketplace.
- The government needs to let good software contractors make money.
- The government needs to make Ada excellence a competitive advantage.
- The DoD mandate is not achieving its intended objectives. It is too flexible and doesn't have teeth.
- The business case supporting the use of Ada is not well constructed and has not been communicated well to industry executives. There are not enough global statistics to substantiate Ada's excellence.
- Ada has not been well marketed.
- A continuing need exists for a broader-based education and training program for all levels of government and industry personnel.
- Availability of quality tools, bindings, and COTS products continues to be an impediment to adoption.

RECOMMENDATIONS (You Do)

In order to strengthen the mandate, the following actions are needed:

- Eliminate wavier process for mission-critical systems (close the door).
- Reinforce DoD language policy to include high integrity, R&D, and information systems where life cycle cost issues are paramount.
- Provide incentives in hiring during source selections for those firms who have proven Ada capabilities and who practice an Ada mandate within their own internal software efforts.
- Provide awards for DoD Program Managers and firms who demonstrate Ada competence during contract performance.
- Have IG investigate compliance of major programs to public law and good engineering practices.
- Refocus the Software Engineering Institute's programs to Ada.

In order to develop a business case, the following actions are needed:

- Capture success stories and lessons learned. Focus on why Ada was selected and what benefits resulted.
- Develop quantitative data on comparative performance (cost, schedule, quality, etc.) of Ada on a life cycle basis in cooperation with the proposed National Repository (Mosemann initiative).
- Cooperate with existing efforts (SIGAda, Ada Software Alliance, etc.) to develop a business case white paper and briefing.

In order to better market Ada, the following actions are needed:

- Define the market and market needs for Ada within the DoD. (Remember the concern over not making this a popularity contest.)
- Improve the outreach by establishing a marketing objective for the AJPO and a plan to realize it - reach out to professional societies, FFRDCs, trade associations, other government agencies, universities, and industry.
- Establish an executive board to act as a sounding board for ideas (reconstitute the Ada board). Make sure the people needed to move Ada from a technology-driven to a market-driven strategy are on it.

In order to provide incentives, the following actions are needed be considered:

- Permit training and tool costs to be directly charged to the contract.
- Increase the Ada competence of the acquisition staff (including their FFRDCs and support contractors).
- Provide advantages in competitive evaluations for Ada excellence.
- Provide incentives in award fee and incentive fee contracts for the use of Ada.
- Permit software development environment assets to be carried as capital equipment to promote (rather than inhibit) investment in quality environments such as Ada tools.

In order to fill the continuing need for broad-based education and training, the following actions are needed.

- Provide grants and other forms of assistance to high schools, universities, and colleges for Ada-based software engineering training.
- Provide a teaching environment and low-cost compilers (free, if possible) for Ada-based software engineering training.
- Provide preference in hiring to Ada trained software entry level engineers and programmers.
- Provide Ada 9X training courseware to firms for use in their continuing education programs.
- Provide videos and Multi-media based educational software for self-paced management education.
- Direct Service schools (DSMC, etc.) and academies to offer Ada-based software engineering/management curriculum.

In order to provide necessary tools, bindings, and COTS products, the following actions are needed:

- Perform a market survey to define the high return investments.
- Enter into partnerships (CRADA's, TRP's, etc.) to jointly develop high demand items on a cost-sharing basis (government takes limited rights, contractor gets marketing rights).

- Use reuse libraries as distribution mechanism for high demand items on a fee-for-service basis.

- Prove value by piloting use of items before release to the open market.

3.0 QUESTION #2: What investments can we make to increase the commercial use of ada?

FINDINGS (We Heard)

The following findings were the result of participant briefings:

- Tools and bindings (WINDOWS/PC first) were noted as the investments with some of the highest potential return.

- The availability of Ada 9X tools, bindings and implementation support needs to be accelerated to maintain the aura of currency.

- Education and training subsidies/grants to colleges and universities who teach Ada and use it in their research seem warranted.

- Strategic partnerships (with other government agencies, firms and universities) should be pursued.

- Wide exposure/marketing investments are needed to change image and create market push.

- Research is needed to resolve Ada 9X technical issues and stimulate universities to teach Ada in both their graduate and undergraduate curriculum.

- The development of Ada 9X superset/subset for education needs to be supported.

- Financial incentives for businesses are needed to motivate them to renew their Ada initiatives.

- Investments in ASIS (Ada Semantic Interface Specification) seem warranted in order to provide the means to interface Ada language and CASE tools through a common repository.

- Reuse needs to be incentivized by resolving the management issues which act as the major inhibitors (data rights, infrastructure costs, organizational issues, etc.).

- Sponsorship of true dual-use programs are needed to enable small businesses to take DoD technology and build products which can be sold broadly to the private sector.

- Giving job preference to those with Ada skills should be considered.

- SBIR (Small Business Innovative Research) programs should be employed to stimulate innovative ideas for Ada commercialization.
- A professional market research firm should be hired to understand the commercial marketplace and develop strategies for effective penetration.
- The entry costs for Ada for educators and their students (costs for compilers, courseware development, etc.) should be lowered.
- The development of a low cost, production-level Ada teaching environment for educators should be subsized.
- Sharing data rights with firms to lower costs for software development should be considered.

CONCLUSIONS (We Think)

Based on the session findings, the following conclusions were made by the panelists:

- Increased DoD investments are necessary to achieve higher degrees of commercial use.
- The following investments seem to have potential high payoffs:

  - Bindings (interfaces to standards like X-Windows, databases and generators)
  - COTS products (standard packages like math libraries, etc.)
  - Marketing/Promotion
  - Tools (debuggers, linkers/loaders, design packages, etc.)
  - Education and Training
  - Trade shows

- A broader buy-in to Ada will help the DoD leverage its investments and achieve broad-based success.
- The DoD and commercial markets are different; do not force Ada in system areas that are not appropriate for Ada.

RECOMMENDATIONS (You Do)

The DoD should ensure there is an Ada market by solidifying the existing base and viewing migration systems within DISA as a potential opportunity for showcasing Ada.

To develop a viable Ada investment strategy, the following actions are needed:

- Take the results of the dual-use workshop and develop a prioritized plan which lays out tasks, schedules, budgets and deliverables incrementally.
- Acquire funding and go do it.
- Provide quarterly feedback and reporting to both ASD (C3I) and DISA, and an executive-level advisory board (company presidents and CEOs, university deans, and department chairs, etc.).

The DoD should establish joint investments/strategic partnerships with:

- Other government agencies (DOT/FAA, NASA, DOE, NIST, etc.)
- Universities/colleges
- Industry/professional societies/trade associations
- Dual-Use programs (NIST, ARPA, NIST, etc.)

The DoD should make reuse commercially viable by opening the reuse libraries and stimulating technology transfer using the dual-use initiative.

4.0 QUESTION #3: What can we do to increase ada's appeal, adoption a nd popularity within your software community?

FINDINGS (We Heard)

The following findings were the result of participant briefings:

- Make sure there are jobs for Ada professionals.
- Institute cooperative initiatives with professional societies, trade associations, etc.
- Stimulate research into Ada issues.
- Provide more widespread exposure to the reasons why Ada was developed and what it can do for firms adopting it.
- Expand the use of Ada into other government activities.
- Change accounting rules (FASB) to stimulate investments in Ada

- Make Ada even more cost-effective (price/performance, ROI, etc.).

- Insert Ada into education process earlier (high school, undergraduate, etc.).

- Counter negative perceptions through an aggressive public relations campaign.

- Try to speed cycle time and make the Ada standardization process more responsive to change.

- Make Ada more attractive by integrating it with the DoD reuse initiative.

- Provide inexpensive tools and compilers.

- Reconstitute the Ada Board as an executive advisory board.

- Foster consistency by providing positive communications.

CONCLUSIONS (We Think)

Based on the session findings, the following conclusions were made by the panelists:

- There is a "second" window of opportunity for Ada stimulated by:

  - Megaprogramming
  - Object-oriented technology
  - High integrity systems
  - Distributed systems client-server architectures

- We need to widen the market base outside of the DoD niche, but we must realize that it is not for every market.

- We need to encourage research institutions to employ Ada as a core technology.

- We need to encourage investment in software by changing the accounting rules and rights in software clause of the FAR

RECOMMENDATIONS (You Do)

In order to take advantage of the "2nd window of opportunity" the following actions are needed:

- Put more business focus on Ada plans and strategies

- Accelerate the migrations to Ada 9X by providing access to quality tools, bindings, training, transition support, and implementation guidance within the next 6 months.

- Promote Ada as a leading edge technology.

- Manage the opportunity and take full advantage of it by leveraging lessons learned in the early years of Ada.

In order to widen the market, the following actions are needed:

- Hold a summit with other agencies to define cooperative strategies which benefit everyone.

- Negotiate MOA's and leverage the government's purchasing power to create demand.

- Create joint acquisition/advisory board to improve communications and liaison with key acquisition officials.

In order to encourage research, the following actions are suggested:

- Have a few Principal Investigators examine potential issues in Ada-based products.

- Pursue creation of subsets and supersets for teaching and research purposes.

- Make it easy for entrepreneurs to access technology necessary to innovate and build new products.

5.0 PANEL

Panel members were selected to represent various facets of the industrial community.

	Name	Organization
Panel Chair:	Dr. Terry Straeter	GDE Systems, Incorporated
Panel Facilitators:	Mr. Don Reifer	Institute for Defense Analyses
	Ms. Debbie Dowling	Electronic Data Systems
Panel Members:	Mr. Milton Cooper	Computer Sciences Corporation
	Mr. John Marciniak	CTA, Incorporated
	Mr. James McCall	SAIC
	Mr. Walker Royce	TRW

6.0 PARTICIPANTS

Participants were allocated a maximum of 15 minutes to give panel members their comments to questions within the industry discipline. The following participants presented briefings during the Ada Dual-Use Workshop (Refer to Section 7.0 of this Appendix, Participant Briefings for copies of presentations). Participants are listed in the order of presentation.

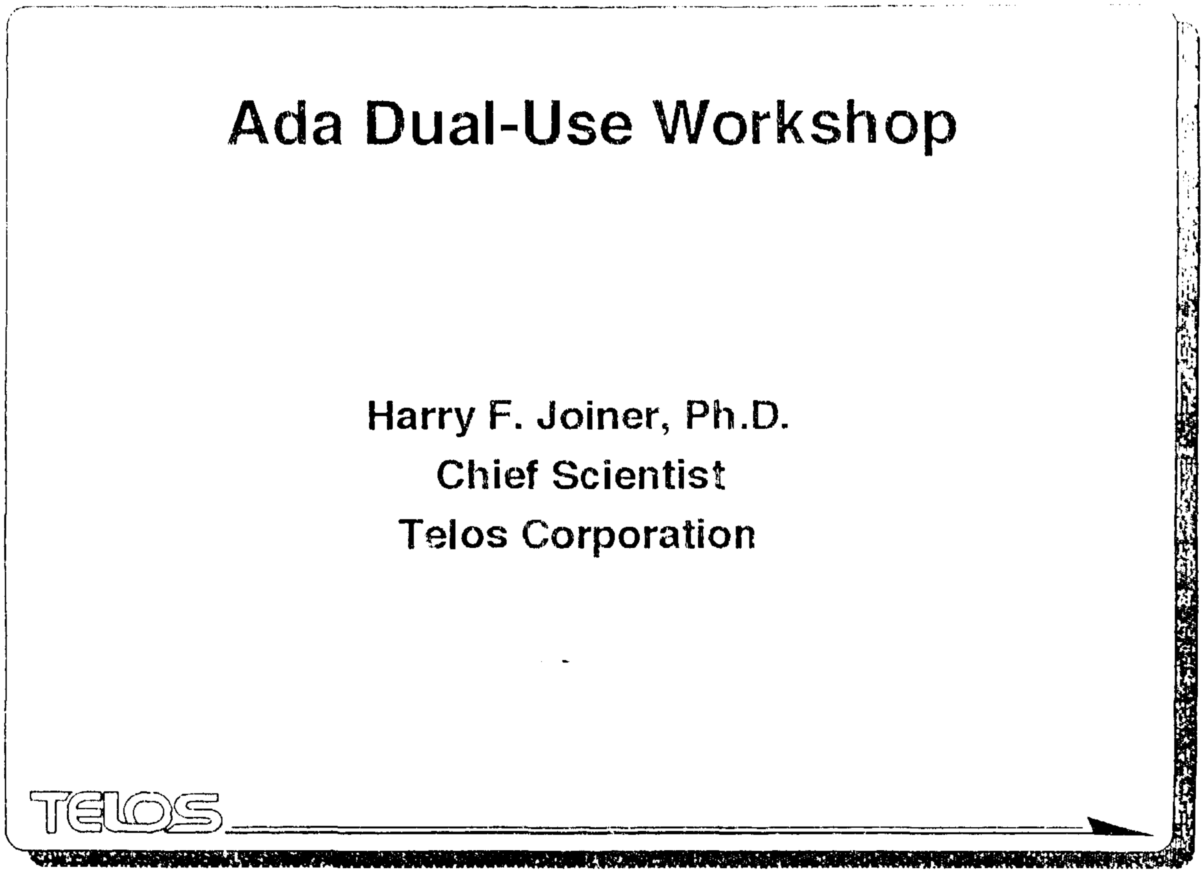
Name	Organization	Briefing
Joiner, Harry	Telos Corporation	Presentation - Slides
Dikel, David	Applied Expertise, Inc.	Presentation - Slides
Balfour, Brad	SofTech, Inc.	Presentation - Slides/PositionPaper
Schwarm, Stephen	The Mitre Corporation	Presentation - Slides
Waterman, Deb	Statistica, Inc.	Presentation - No Slides
Aharonian, Gregory	Source, Translation and Optimization	Presentation - Position Paper
Hart, Hal	TRW	Presentation - No slides
Stevens, Scott	Magnavox	Presentation - Slides/Position Paper
Krell, Bruce	Hughes	Presentation - Slides
Kennedy, Gary	IBM Federal Systems Corp.	Presentation - Slides/Position Paper
Leif, Robert	Ada-Med	Presentation - Slides
Crafts, Ralph	Ada Software Alliance	Presentation - Slides
Kester, Rush	DC SIGAda	Presentation - Slides
Henry, John	EDSI	Presentation - Slides
Riehle, Richard	AdaWorks	Presentation - Slides
Maymir-Ducharme, Fred	Unisys	Presentation - Slides
Johnson, Barbara	Raytheon	Presentation - Slides
Kidwai, Shakil	Electronic Data Systems	Presentation - Slides
Nyberg, Karl	Grebyn Corporation	Presentation - Slides

Note: Some of the views expressed were solely the participant's own and not of his/her organization.

7.0 PARTICIPANT BRIEFINGS

This section contains a copy of participant briefing materials provided to the Industry Panel during the Ada Dual-Use Workshop. In cases were presentation materials were submitted in handwritten form, typed copies were created by the Ada Dual-Use Workshop staff. These copies are noted accordingly.

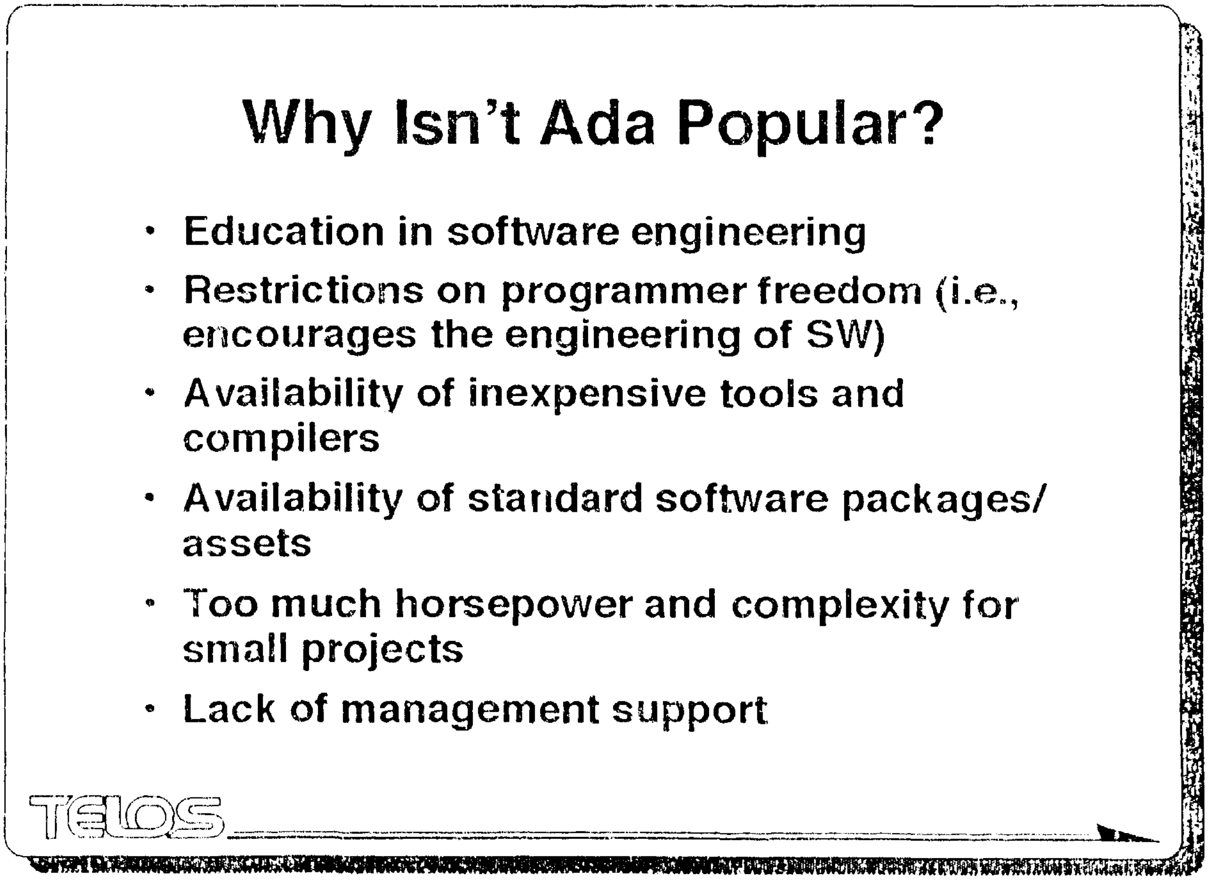
Participant Briefing
Harry Joiner
Telos Corporation



Ada Dual-Use Workshop

Harry F. Joiner, Ph.D.
Chief Scientist
Telos Corporation

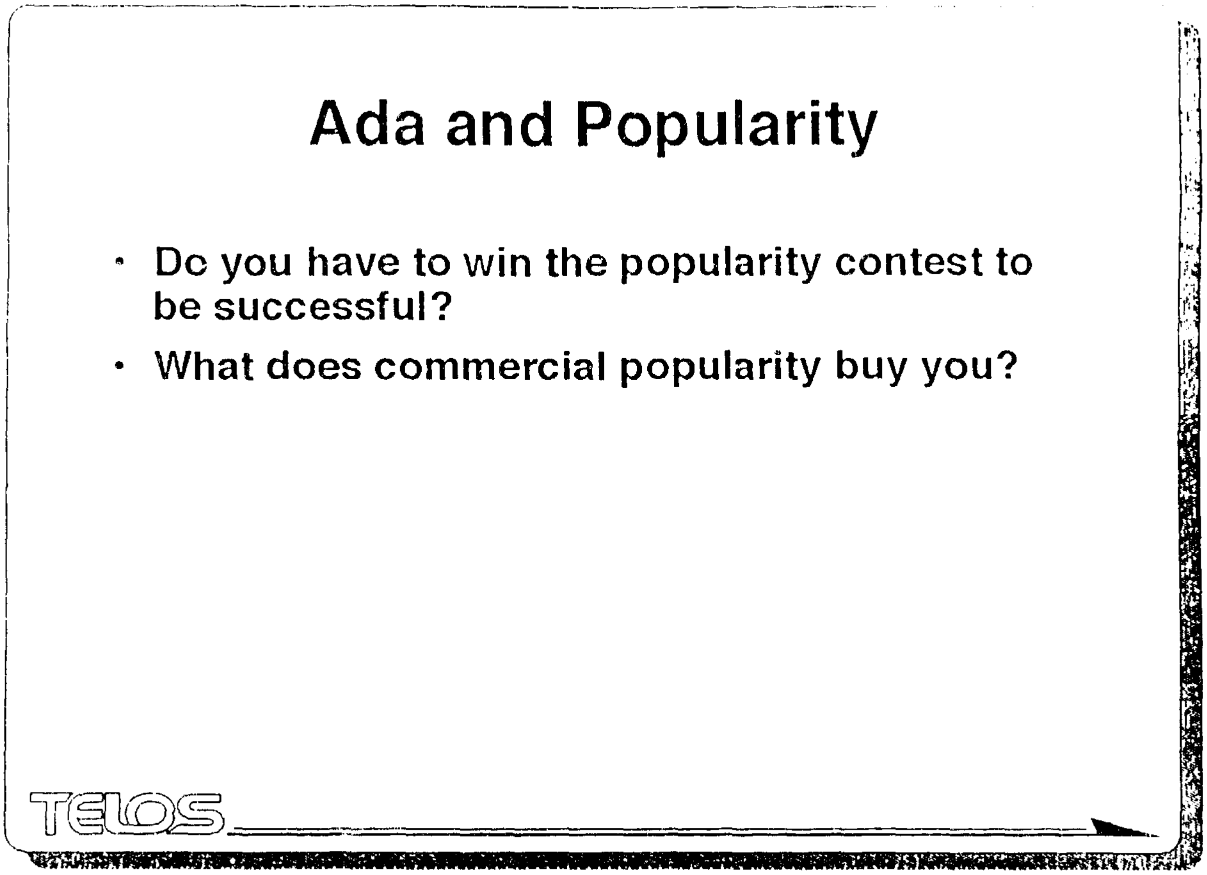
TELOS



Why Isn't Ada Popular?

- Education in software engineering
- Restrictions on programmer freedom (i.e., encourages the engineering of SW)
- Availability of inexpensive tools and compilers
- Availability of standard software packages/ assets
- Too much horsepower and complexity for small projects
- Lack of management support

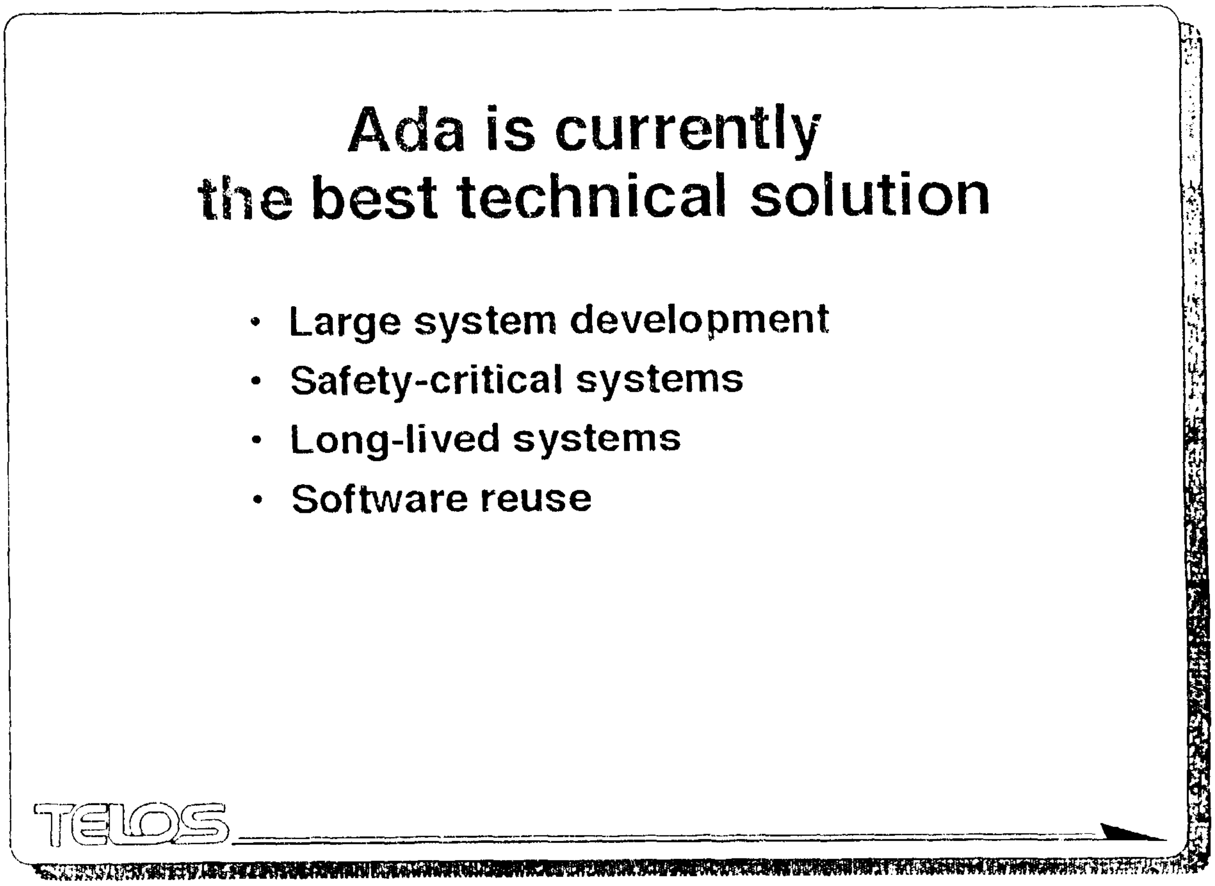
TELOS



Ada and Popularity

- Dc you have to win the popularity contest to be successful?
- What does commercial popularity buy you?

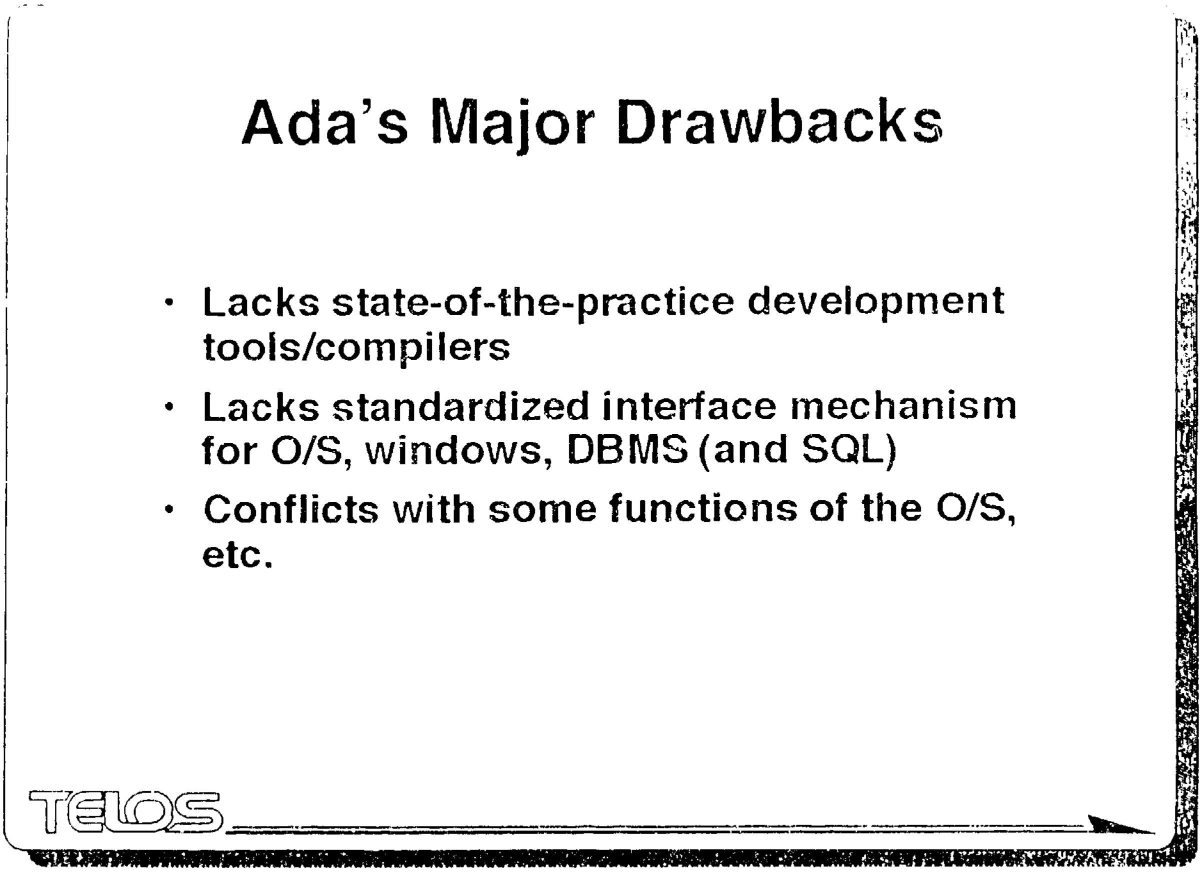
TELOS



Ada is currently the best technical solution

- Large system development
- Safety-critical systems
- Long-lived systems
- Software reuse

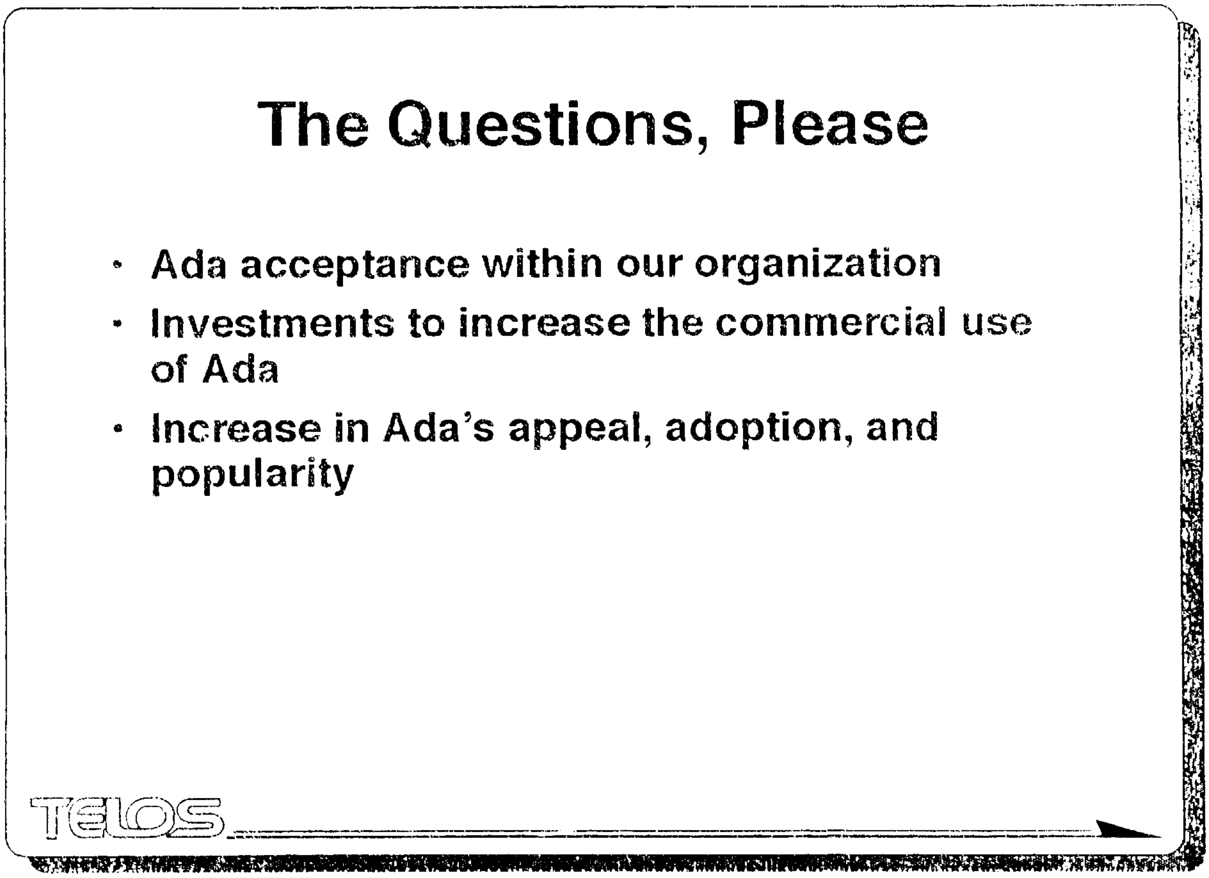
TELOS



Ada's Major Drawbacks

- Lacks state-of-the-practice development tools/compilers
- Lacks standardized interface mechanism for O/S, windows, DBMS (and SQL)
- Conflicts with some functions of the O/S, etc.

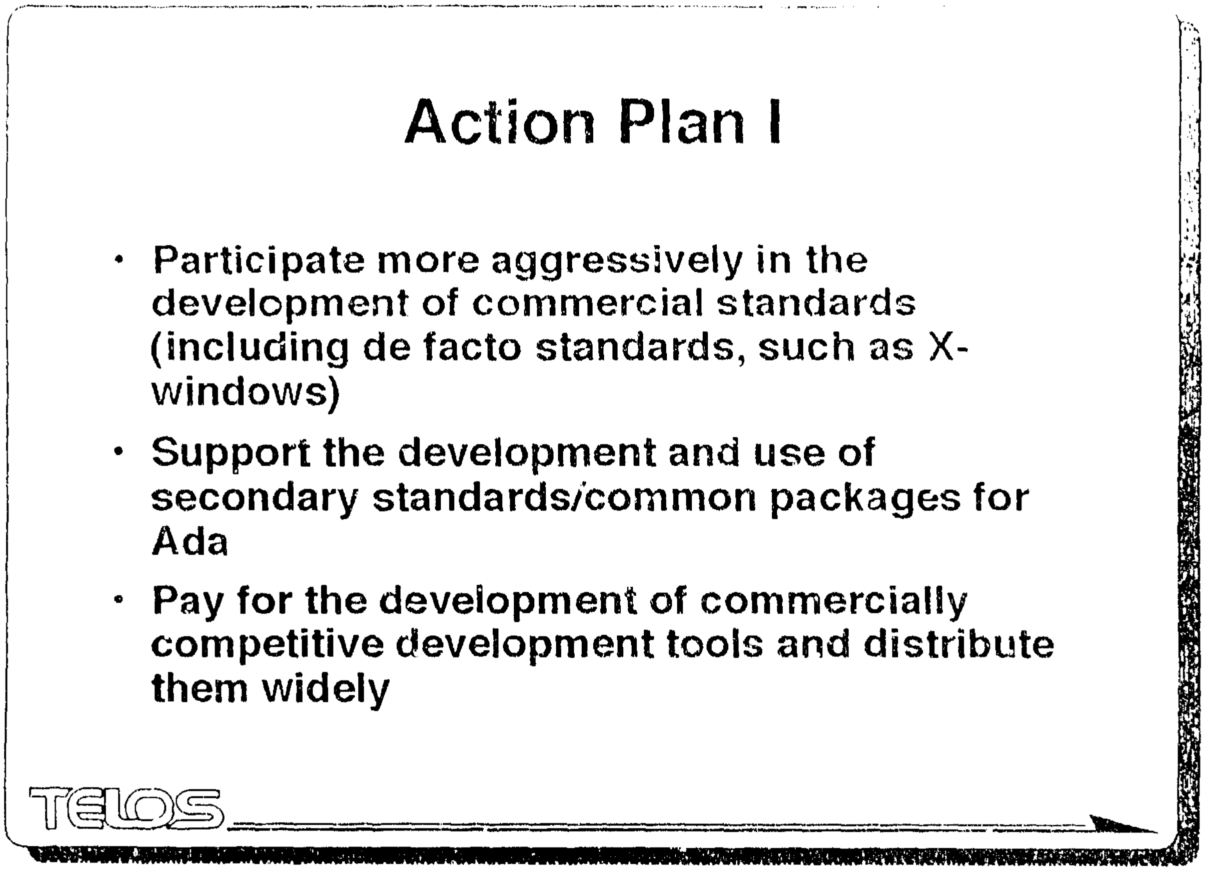
TELOS



The Questions, Please

- Ada acceptance within our organization
- Investments to increase the commercial use of Ada
- Increase in Ada's appeal, adoption, and popularity

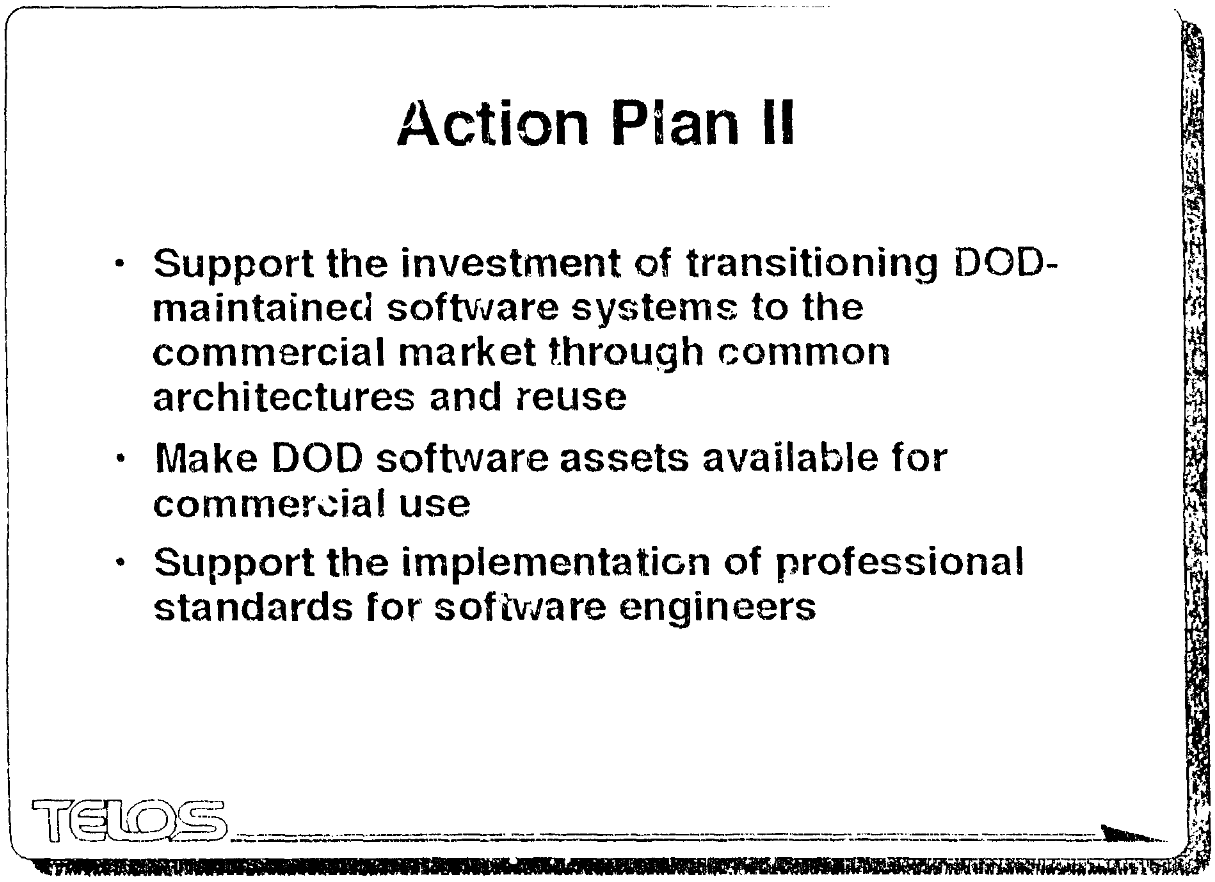
TELOS



Action Plan I

- Participate more aggressively in the development of commercial standards (including de facto standards, such as X-windows)
- Support the development and use of secondary standards/common packages for Ada
- Pay for the development of commercially competitive development tools and distribute them widely

TELOS



Action Plan II

- Support the investment of transitioning DOD-maintained software systems to the commercial market through common architectures and reuse
- Make DOD software assets available for commercial use
- Support the implementation of professional standards for software engineers

TELOS

Participant Briefing

David Dikel
Applied Expertise, Incorporated

Presentation to the Ada Dual-Use Workshop

Vianna, Virginia

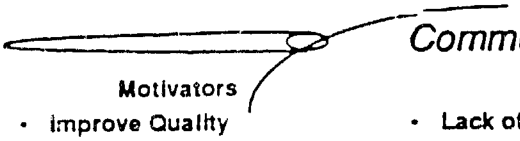
October 19-20, 1993

Dave Dikel
Applied Expertise, Inc.
1925 North Lynn Street, Suite 802
Arlington, VA 22209
703 516-0911
Fax: 703 516-0918
ddikel@ajpo.sei.cmu.edu

Background

- Intellimac [1981...]
  - First commercial Ada application (1981) – Part of team
  - First integrated Ada product – Supported many pioneering Ada applications
- Commercial Ada Users Group [1986...]
  - Founded (1986) under AJPO/IDA sponsorship
  - Focused on questions an industry manager asked
    - Participants included Boeing, GE, Shell, Reuters, Oracle
    - Meetings drew many government managers, anxious to hear answers
- Case studies [1986...] (Reuters, Dowell-Schlumberger, Nokkia, Boeing, GE, Xerox, others)
- Presentations, tutorials [1986...]
- Ada Market Study [1990]
- Specialized user studies on adaptation issues, vendor studies [1989...]

Lessons from the Past



Common Threads

Motivators

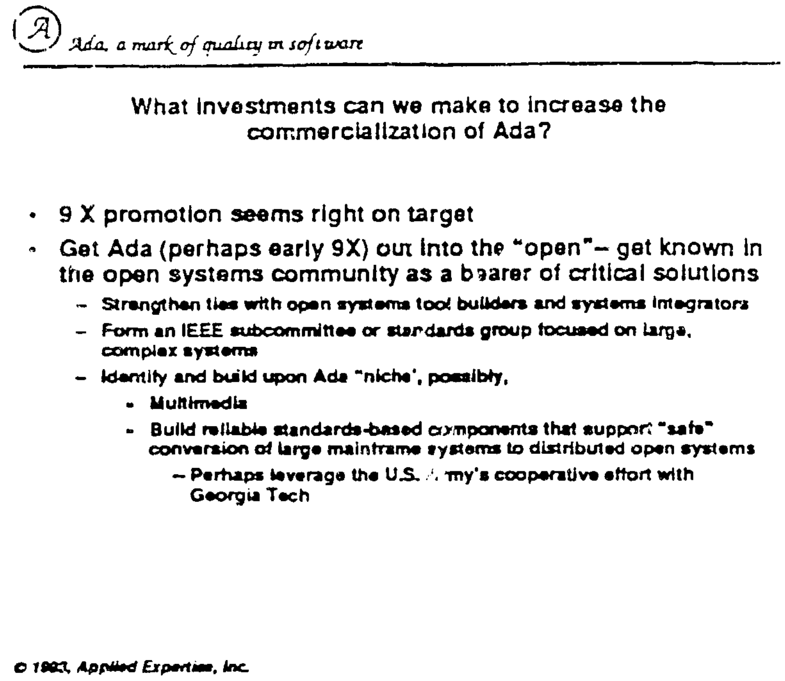
- Improve Quality
- Increase reuse
- Introduce software engineering
- Mandate...
  - Products, methods, people proven on M + LOC systems
  - Advocates that have experience to show gains in quality, productivity

Barriers

- Lack of interfaces to commercial standards
  - O/S - other tool builders don't use much Ada
  - Extra time and $ required
- ...Mandate
  - Expensive products/people for "captive audience"
  - Reaction to forced use/ military image... "It couldn't be any good"

What actions can we take to get Ada accepted as the preferred language within your organization ?

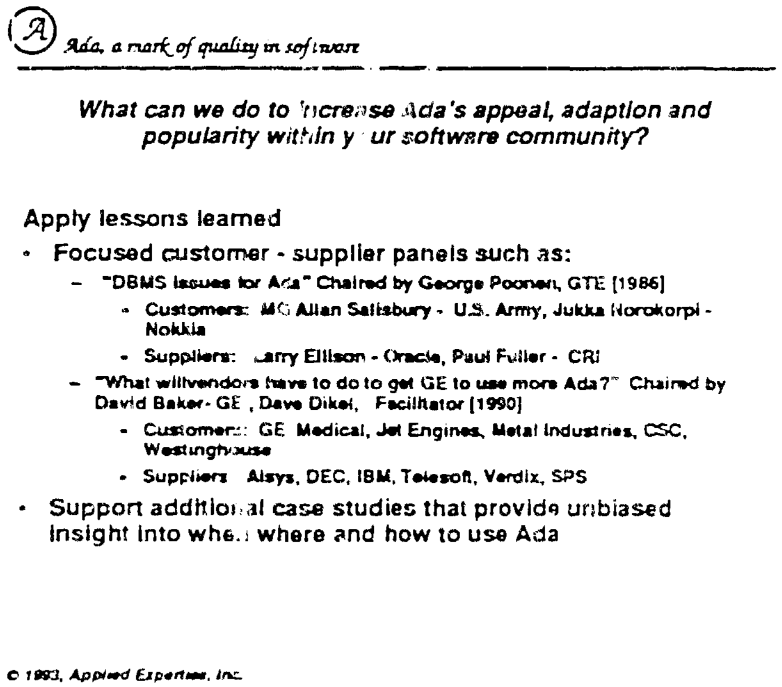
- Background: AE builds information systems and tools such as:
  - CBD Express – A widely used commercial product that automates the preparation and transmission of Commerce Business Daily announcements
  - Case tracking information system for the National Mediation Board – A central resource for scheduling and recording NMB's activities
  - Contacts management database for the International Finance Corporation (World Bank)
- Suggestion: (Not necessarily easy) Encourage Borland and other database vendors to:
  - Use Ada themselves
  - Provide tools that support Ada that are as good or better than tools for other languages – e.g. Ada equivalent to the "C Engine"



Ada, a mark of quality in software

What investments can we make to increase the commercialization of Ada?

- 9 X promotion seems right on target
- Get Ada (perhaps early 9X) out into the "open"– get known in the open systems community as a bearer of critical solutions
  - Strengthen ties with open systems tool builders and systems integrators
  - Form an IEEE subcommittee or standards group focused on large, complex systems
  - Identify and build upon Ada "niche', possibly,
    - Multimedia
    - Build reliable standards-based components that support "safe" conversion of large mainframe systems to distributed open systems
      - Perhaps leverage the U.S. A my's cooperative effort with Georgia Tech



Ada, a mark of quality in software

What can we do to increase Ada's appeal, adaption and popularity within y ur software community?

Apply lessons learned

- Focused customer - supplier panels such as:
  - "DBMS Issues for Ada" Chaired by George Poonen, GTE [1986]
    - Customers: MG Allan Saltsbury - U.S. Army, Jukka Norokorpi - Nokkia
    - Suppliers: Larry Ellison - Oracle, Paul Fuller - CRI
  - "What willvendors have to do to get GE to use more Ada?" Chaired by David Baker- GE , Dave Dikel, Facilitator [1990]
    - Customers: GE Medical, Jet Engines, Metal Industries, CSC, Westinghouse
    - Suppliers Alsys, DEC, IBM, Telesoft, Verdix, SPS
- Support additional case studies that provide unbiased insight into whe. where and how to use Ada

Participant Briefing
Brad Balfour
SofTech, Incorporated

SOFTECH

Ada Dual-Use Workshop

SofTech Position Paper
How DoD Can "Revitalize" Ada

Brad Balfour

SOFTECH
1600 N. Beauregard St.
Alexandria, VA 22311
voice: (703) 824-4505
fax: (703) 931-6530
email: bbalfour@softech.com

SOFTECH

"Revitalize Ada" — What Does It Mean

- ❑ Combatting a *perception*
  - ○ That Ada isn't successful
  - ○ That Ada isn't growing
  - ○ That Ada isn't a viable technical solution
- ❑ DoD alone can't do it
- ❑ DoD in partnership with industry, academia, other Federal Agencies, professional societies, and trade groups can do it

SOFTECH

5 Key Activities

1. Reconfirm DoD's Commitment to Ada
2. Collect and Disseminate Ada's Successes
3. Focus DoD and CIM Efforts on Ada
4. Promote Industry Dual Usage of Ada
5. Promote Ada 9X and Ada 9X Education

SOFTECH

This is the most important action that DoD can take

Reconfirm DoD's Commitment to Ada

- ❑ Advertise the fact that Ada is DoD's corporate standard and has been chosen for good reasons
- ❑ Remember that programmers view languages as an almost religious issue
- ❑ Make an analogy to the other corporations that standardize on a single language, environment, platform or tool set
- ❑ Enforce the use of Ada on all types of systems
  - ○ Minimize exceptions
  - ○ Present a *single positive message* in public
  - ○ Ensure that ARPA, NSA, MITRE, etc. are visibly using Ada

Collect and Disseminate Ada's Successes

- □ Several steps to accomplish this:
  - ○ Policy established requiring all projects within DoD to collect and report several simple kinds of information
  - ○ Funding and expenditure reports on DoD development activities can be collected by AJPO and AdaIC
  - ○ Business case information as well as Ada usage and savings information can then be developed and disseminated
- □ DoD must then collect Ada project information and write up "Ada success stories
  - ○ Present as papers at both Ada and non-Ada conferences
  - ○ Also disseminate via:
    - ◆ Press releases to newspapers / newsletters
    - ◆ Articles submitted to conferences and magazines
    - ◆ Postings to electronic bulletin boards and mailing lists
    - ◆ Advertisements in journals and newspapers
    - ◆ Packets with detailed success stories available from AdaIC
    - ◆ Booths at conferences and trade shows

Focus DoD and CIM Efforts on Ada

- □ Reuse Repositories
  - ○ Already focus on Ada RSCs as a core technology
  - ○ Narrow their focus to just Ada for code components
- □ Center for Standards
  - ○ Supplement Technical Reference Model with available bindings
  - ○ If bindings don't exist, subsidize efforts to develop them and place in public domain
  - ○ Include efforts to submit them for standardization by ANSI or ISO
  - ○ Include "defacto" standards
  - ○ Provide a presence on both formal standards committees and industry coordination groups setting "defacto" standards
    - ◆ Offset the heavy C and C++ bias of these groups
    - ◆ Ensure Ada APIs are defined at the same time the C/C++ APIs are defined

Promote Industry Dual Usage of Ada

- ❑ Dual-Use Incentives
  - ❍ Provide contracting policy and regulations that give incentives to contractors who demonstrate dual-use of Ada within their non-DoD commercial applications
  - ❍ Incentives could take the form of proposal evaluation criteria
  - ❍ Government could provide "dual-use" funds similar to the ATIP program
    - ◆ Could be used to leverage DoD technology into "dual-use" technologies
  - ❍ Follow up these efforts with press coverage

Promote Industry Dual Usage of Ada (2)

- ❑ Accelerate Deployment of Ada Compilers
  - ❍ Use the DoD's leverage as one of the largest customers of Ada tools
  - ❍ Work closely with the vendors to ensure that:
    - ◆ Ada compilers are available simultaneously with the release of new hardware
    - ◆ Ada compilers are priced competitively with C and C++ compilers
    - ◆ "Sidgrade" pricing is available to owners of competing language compilers
    - ◆ A Borland or Microsoft "name brand" Ada compiler is available

Promote Ada 9X and Ada 9X Education

- ❑ Educate the Non-Ada Community on Ada 9X
  - ❍ Joint with SIGAda, academia and trade associations
  - ❍ Target existing OOPL markets including:
    - ◆ Reengineering MIS applications from legacy COBOL to OO Client/Server systems in an OOPL
    - ◆ Organizations thinking of developing new applications using OOP in C++
    - ◆ Distributed computing applications — Client/Server, simulation and supercomputing
    - ◆ Non-DoD embedded systems: cars, household appliances, medical instruments, etc.

Promote Ada 9X Education (2)

- ❑ Create Promotional Material
  - ❍ Similar to the way the the DoD has promoted weapon systems in the past
  - ❍ Distributed widely through a variety of sources including AdaIC, conferences, industrial trainers, university educators
  - ❍ Available to anyone at no cost
- ❑ Fund Appearances at Non-Ada Trade Shows
  - ❍ Use AdaIC in conjunction with ACM SIGAda and industry trade groups
  - ❍ Promote Ada to new markets
  - ❍ Show a significant presence at non-Ada conferences both nationally and internationally
  - ❍ Provide new developers with information on Ada
    - ◆ Give them enough material to take home and convince their boss
  - ❍ Target many difference conferences

SOFTeCH

SofTech, Inc.

1600 North Beauregard Street
Alexandria, VA
22311-1794

703 931-[illegible]
FAX 703-931-[illegible]

October 19, 1993

Ada Dual Use Workshop Position Paper
How DoD Can "Revitalize" Ada

1. Introduction

There are many things that the Department of Defense — through DISA/JIEO/CIM — can do to "revitalize" Ada. In this context, "revitalize" means combating the *perception* that Ada isn't successful, isn't growing, and isn't a viable technical solution. DoD, alone, cannot "revitalize" Ada, however, in a partnership with industry, academia and other Federal Agencies, DoD can effectively expand Ada's visibility and use.

2. Key Activities

Five specific DoD Ada revitalization activities are recommended below. These activities address ways in which the DoD can build upon its current leadership role in the Ada world and revitalize Ada throughout the software engineering community. The five activities are:

A. Reconfirm DoD's Commitment to Ada

This is the most important action that DISA and the DoD can undertake. Expand opportunities to advertise the fact that Ada **is** DoD's corporate standard and that it has been chosen for very good reasons. Many programmers view languages as an almost religious issue and attempts at standardization and uniformity will encounter resistance. To reduce costs, many corporations standardize on one language for corporate wide use --- this is an appropriate and accepted fact.

DoD must continue to aggressively enforce the Ada standard on all types of systems and minimize the exceptions. Get the support of ARPA, NSA, MITRE and others. Additionally, DoD must place great emphasis on presenting a single, consistent, positive message in all public forums — that Ada is the DoD's choice for software development and is a successful positive technology.

B. Collect and Disseminate Ada's Successes

Ensure that success stories are collected and disseminated. To accomplish this, several things can be done:

(1) Policy can be established requiring *all* software projects within DoD to collect and report several, *simple*, kinds of information (e.g. size of the project, number of developers, length of the project, outcome, etc.). Funding and expenditure reports on DoD software development activities also can be collected by DISA (more specifically to the AJPO and AdaIC). Business case studies for Ada usage and savings can then be developed and disseminated.

(2) DoD must collect appropriate Ada project information and write "Ada success story" professional papers for presentation at Ada and non-Ada conferences. Ada success stories also can be disseminated using several media:

a) press releases to newspapers and newsletters,
b) articles submitted to conferences and magazines,
c) postings to electronic mailing lists and bulletin boards,
d) advertisements in journals and newspapers,
e) packets of detailed project success stories available from the AdaIC,
f) booths at conferences and trade-shows.

This effort to document DoD Ada use must be supplemented with the collection of non-DoD statistics as well. Currently industry, academia and organizations such as SIGAda do this on a volunteer basis; DoD can supplement its volunteer effort with appropriate incentives.

Create additional positive press for Ada. DISA can support the creation of both press releases and articles on Ada. These should be targeted at magazines and newspapers that have not typically had heavy Ada coverage. These would include: Computerworld, Government Computer News, Datamation, IEEE Software, Journal of Object Oriented Programming, etc. DISA should coordinate the release and placement of these press releases and articles with the Ada vendor community so that the vendors can place ads for products in the same magazines at the same time as news articles appear. These articles need not (and perhaps should not) be written by the DoD. However DISA can provide additional funds to successful Ada projects to document their results in the form of articles submitted to (traditionally) non-Ada magazines and conferences. Close cooperation with Industry trade groups and ACM SIGAda is necessary to support this activity.

C. Focus DoD and CIM Efforts on Ada

Ensure that DoD reuse repositories focus on Ada reusable components as a core technology. DISA (through its Center for Standards) can supplement its Technical Reference Model by ensuring that Ada bindings exist for all standards and are freely available through the repositories. For those bindings that do not exist, DISA should subsidize efforts to develop and place them in the public domain. This should include efforts to submit these bindings for standardization by ANSI or ISO. Additionally, DISA should not restrict its efforts to only ANSI or ISO standards, but should work with vendors who supply "defacto" standards to ensure that Ada bindings (APIs) are available for those products as well (e.g. Microsoft Windows, X-windows, etc.)

Provide a presence on industry coordination groups, who over the past five years, have emerged as a powerful factor in setting "defacto standards". DoD and DoD contractor participation in these groups can balance the heavy C and C++ bias and promote Ada requirements, particularly in the Ada API and/or bindings areas. These groups should include, as an example, OMG (which is defining the CORBA and ORB sets of services), the MIT X consortium, OSF, XOpen, COSE (Common Open Systems Environment), CCITT (telecom), X12 (Electronic Data Interchange), SQL Access Group, and ODA Consortium (Open Document Architecture).

D. Promote Industry Dual Usage For Ada

Provide Dual Use Incentives. The DoD can provide contracting policy and regulations that provide incentives to contractors who demonstrate dual-use of Ada within their non-DoD commercial applications. Incentives could take the form of proposal evaluation criteria which promote contractor dual use of Ada in their non-DoD business areas. Additionally, the government could provide "dual-use" funds (similar to the AJPO's highly successful ATIP program) to leverage government Ada investments into "dual-use" technologies. These efforts should be followed up with large amounts of positive press in the form of articles in major newspapers (e.g. Washington Post, Wall Street Journal, etc.) and frequent demonstrations of Ada technology in "dual-use" situations by high ranking government officials.

Accelerate deployment of Ada compilers on strategic platforms. As one of the largest customers of Ada software, the DoD can influence the direction taken by the current Ada compiler vendors. By working closely with the compiler and hardware vendors the DoD can ensure that Ada compilers, like C and C++ compilers, are available *simultaneously* with the release of new hardware platforms. Hardware buys contingent upon the availability of Ada compilers on those platforms also encourages industry cooperation.
DoD also needs to provide a clear message to the vendors that as a customer, it wants compilers that are priced competitively with C++ compilers for the same platform. DoD could also encourage vendors to offer a "sidegrade" from a competing C++ to an Ada compiler to entice people to try Ada. Finally, DoD could approach Borland and Microsoft to provide Ada compilers along with their tools for other languages (even if these compilers are simply OEMd from existing Ada vendors).

E. Promote Ada 9X and Ada 9X Education

Educate the non-Ada community on Ada 9X. Embark on a joint campaign with trade associations, ACM SIGAda, and academia to educate the software/hardware industry on the benefits and features of the new Ada9X. Additionally, target existing object oriented programming languages (OOPL) markets which may provide new niches for Ada:

- Reengineering MIS applications from legacy COBOL to Object Oriented Client/Server systems in OOPL (include reverse/reenginnering tool vendors). Ada9X could be used.

- Organizations who are considering developing their new applications using OOP and C++. Ada9X's object oriented features will be superior.

- Distributed applications including client/server, simulation, and supercomputing. Ada 9X's new distributed computing capabilities are well suited for this area.

- Non-DoD small embedded systems such as the popular embedded controllers found in cars, household appliances, medical instruments, etc. This community shares much with the DoD and could benefit from the many Ada lessons learned.

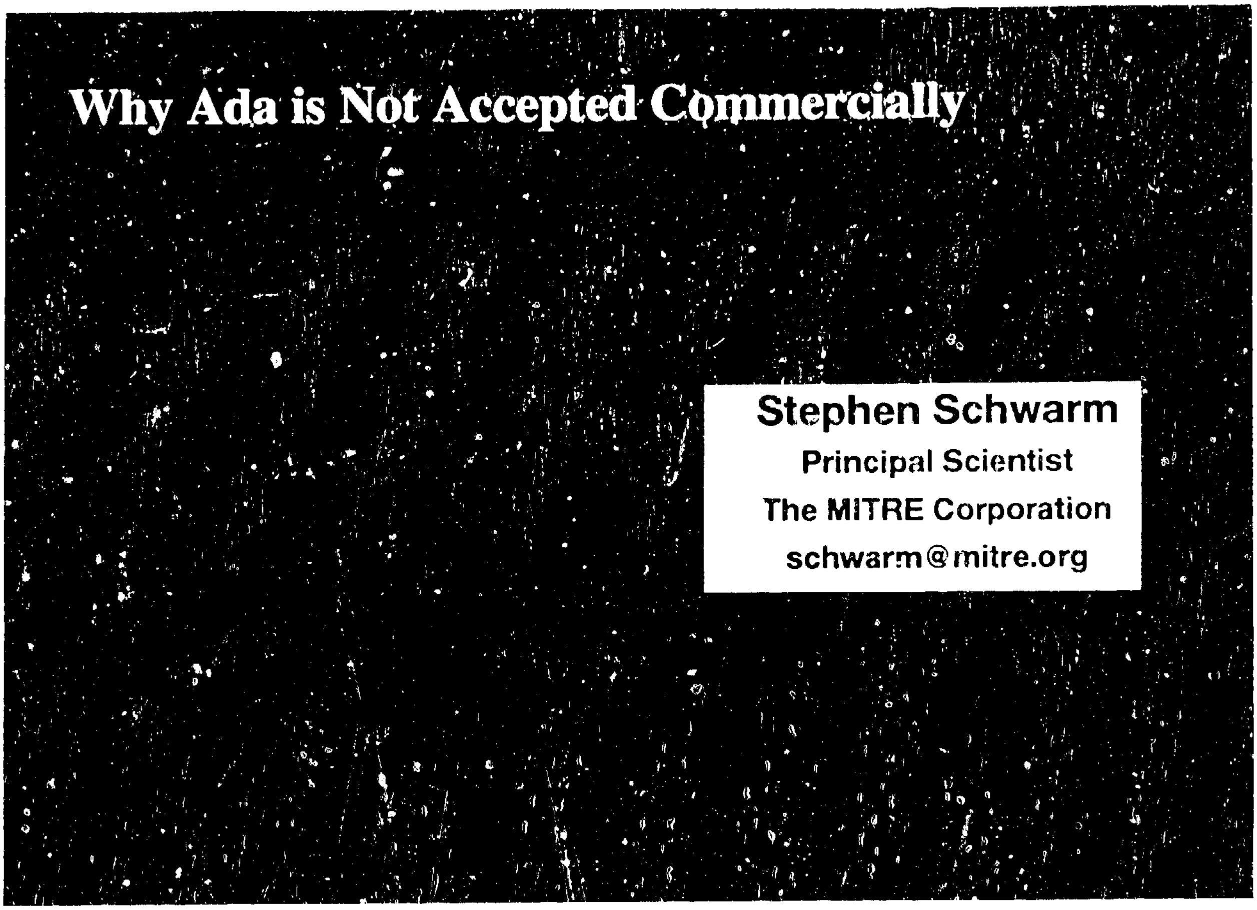
Create promotional material. DISA, using a professional marketing/PR firm, could create promotional material on Ada 9X's features and benefits similar in quality and style used to market large weapons systems in congress. These materials should be distributed widely through AdaIC, Ada vendors, conferences, industrial trainers and university educators at no cost. Materials should appear "generic" to avoid any perception of DoD bias.

Fund appearances at non-Ada trade shows. DoD can and should use the Ada Information Clearinghouse, in joint cooperation with an Ada industry trade-group and with ACM SIGAda, to promote Ada in new markets. Significant presence at non-Ada conferences both nationally and internationally can provide new developers the necessary information to get interested and get started. DoD could initially sponsor an Ada booth represented by a professional organization such as ACM SIGAda. Target major Object Oriented conferences (OOPSLA and Object World), MIS conferences (such as DPMA — the Data Processing Management Association, and ITAA -The Information Technology Association of America), CASE conferences (CASEWorld, etc.), embedded software/real-time/distributed systems conferences (such as Embedded Systems Conference), and major "general" software conferences (such as Comdex).

3. Conclusion

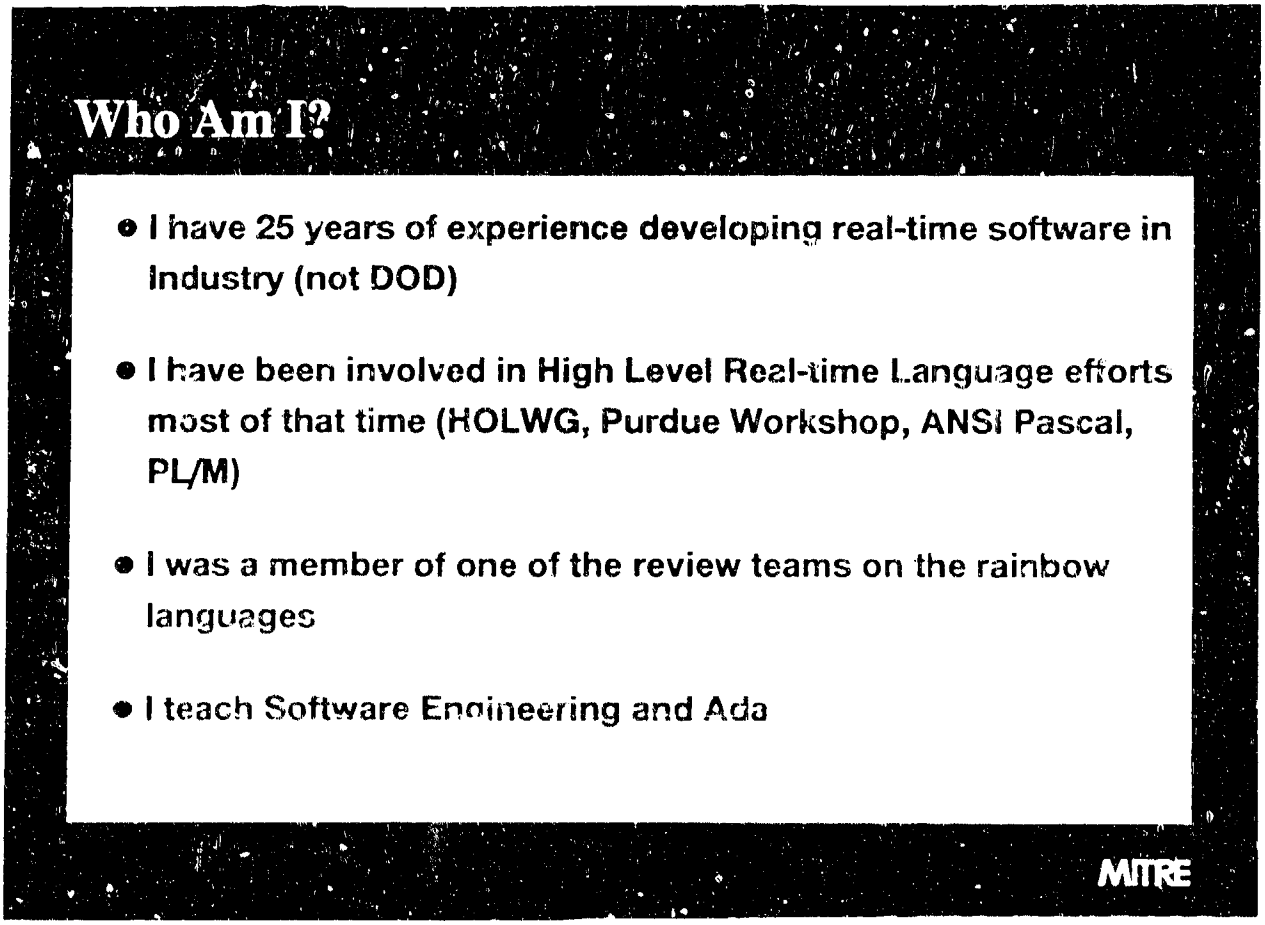
The DoD has a multitude of opportunities to make real progress in efforts to "revitalize" Ada. There are large numbers of DoD workers and support contractors who are knowledgeable and enthusiastic supporters of the language. That is not enough. To achieve strategic long term Ada benefits the DoD needs a solid professional marketing campaign. Marketing, coupled with a focused program of technology insertion incentives and data collection and reporting, will provide the DoD a solid foundation for Ada revitalization throughout the community.

Participant Briefing
Stephen Schwarm
The Mitre Corporation



Why Ada is Not Accepted Commercially

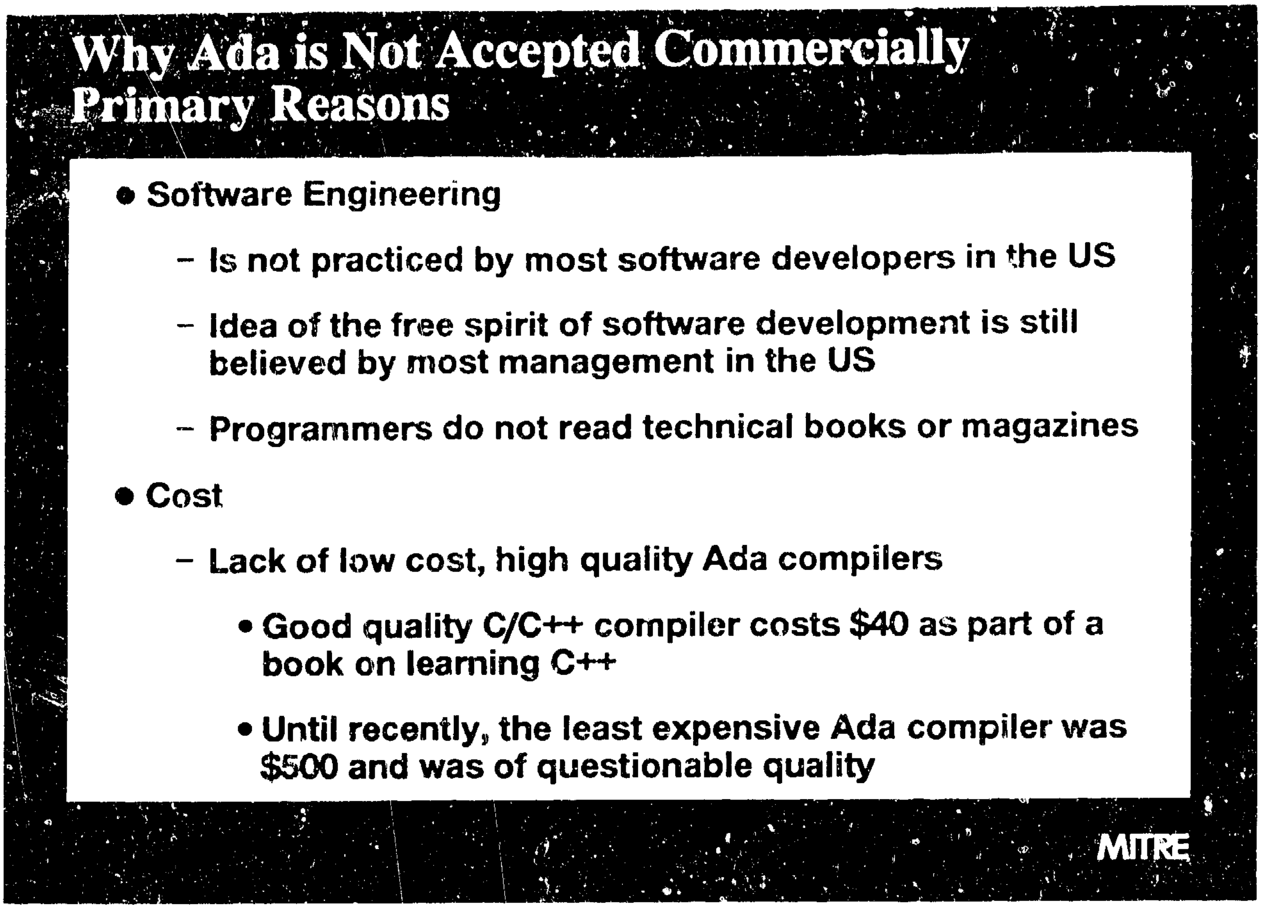
Stephen Schwarm
Principal Scientist
The MITRE Corporation
schwarm@mitre.org



Who Am I?

- I have 25 years of experience developing real-time software in Industry (not DOD)
- I have been involved in High Level Real-time Language efforts most of that time (HOLWG, Purdue Workshop, ANSI Pascal, PL/M)
- I was a member of one of the review teams on the rainbow languages
- I teach Software Engineering and Ada

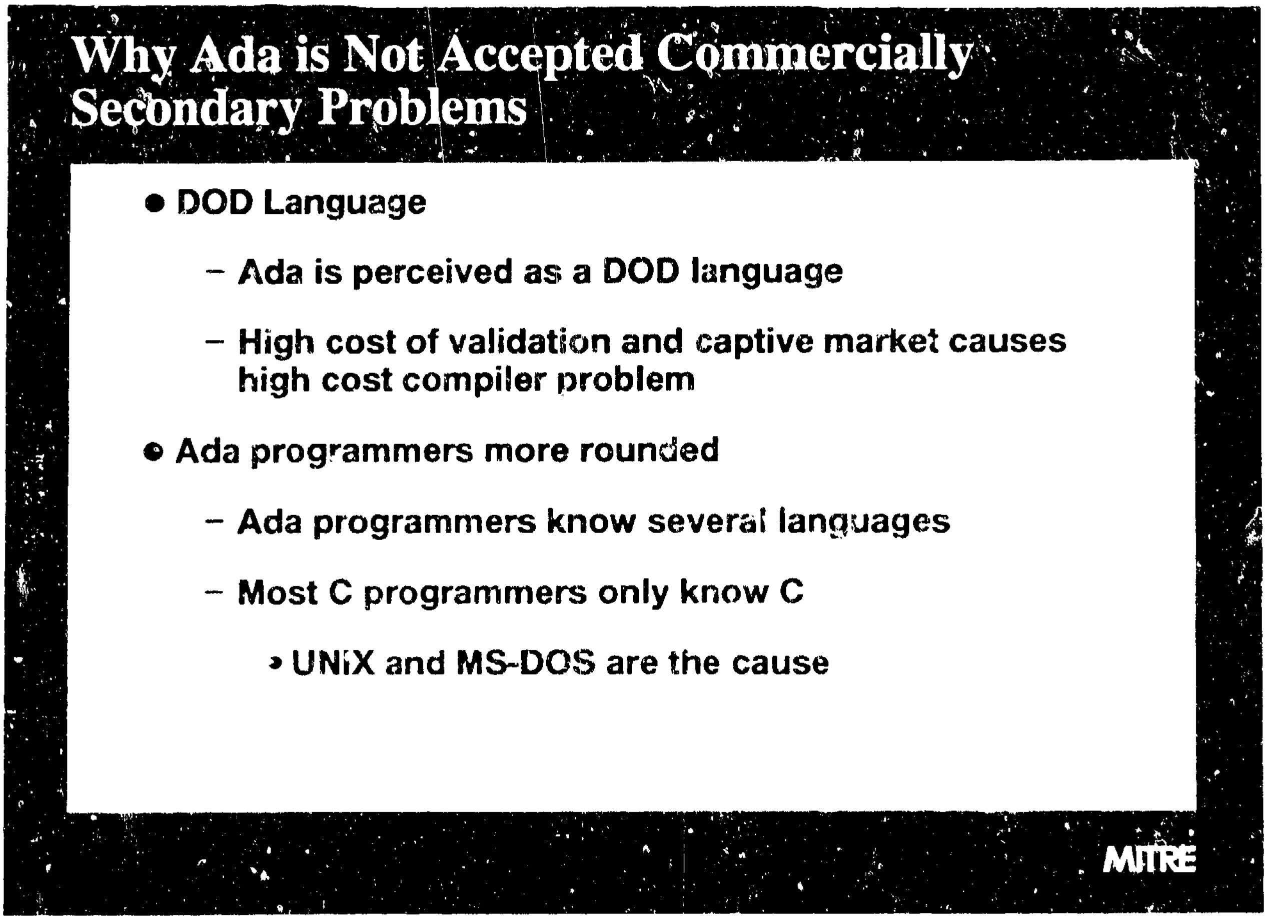
MITRE



Why Ada is Not Accepted Commercially Primary Reasons

- Software Engineering
  - Is not practiced by most software developers in the US
  - Idea of the free spirit of software development is still believed by most management in the US
  - Programmers do not read technical books or magazines
- Cost
  - Lack of low cost, high quality Ada compilers
    - Good quality C/C++ compiler costs $40 as part of a book on learning C++
    - Until recently, the least expensive Ada compiler was $500 and was of questionable quality

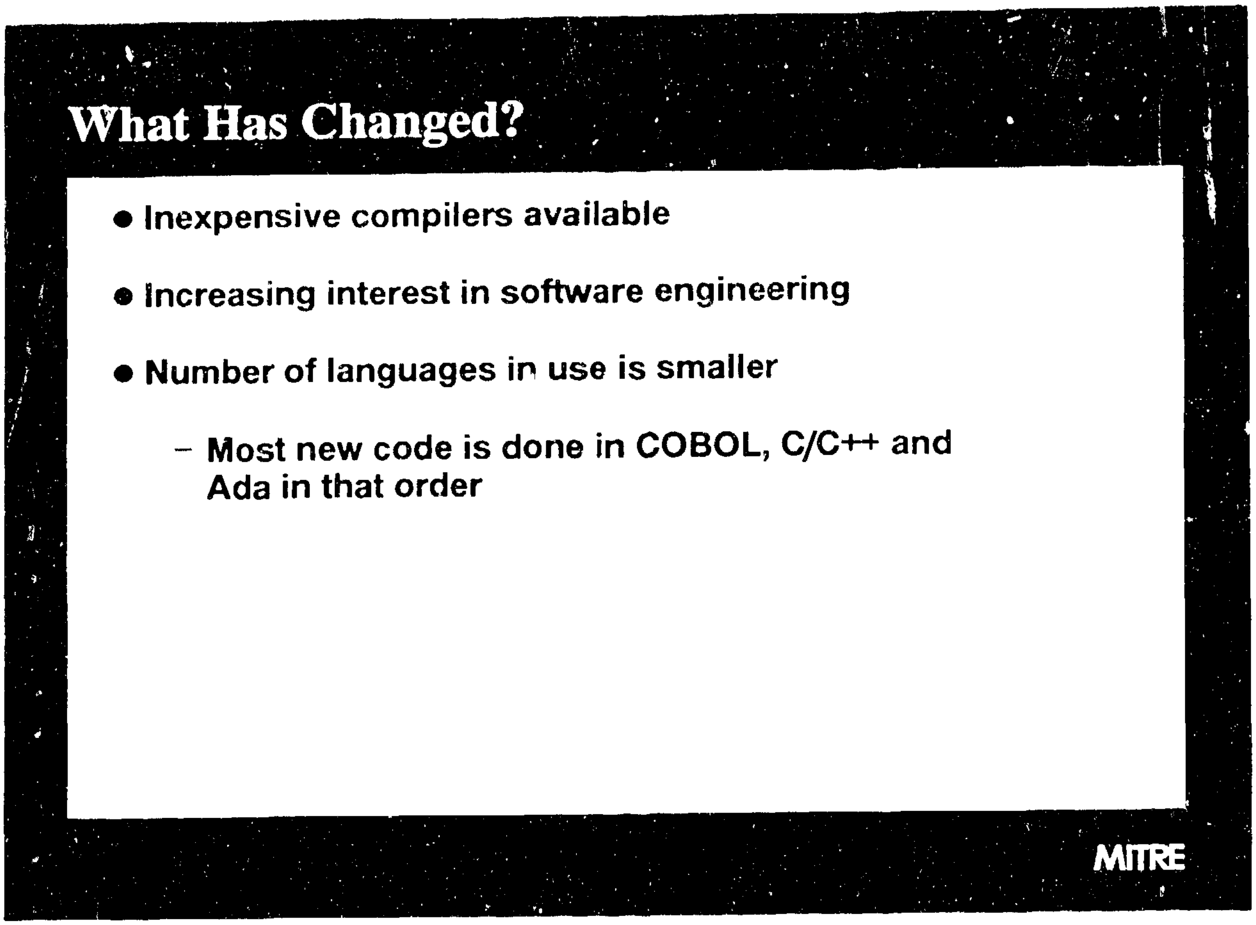
MITRE



Why Ada is Not Accepted Commercially Secondary Problems

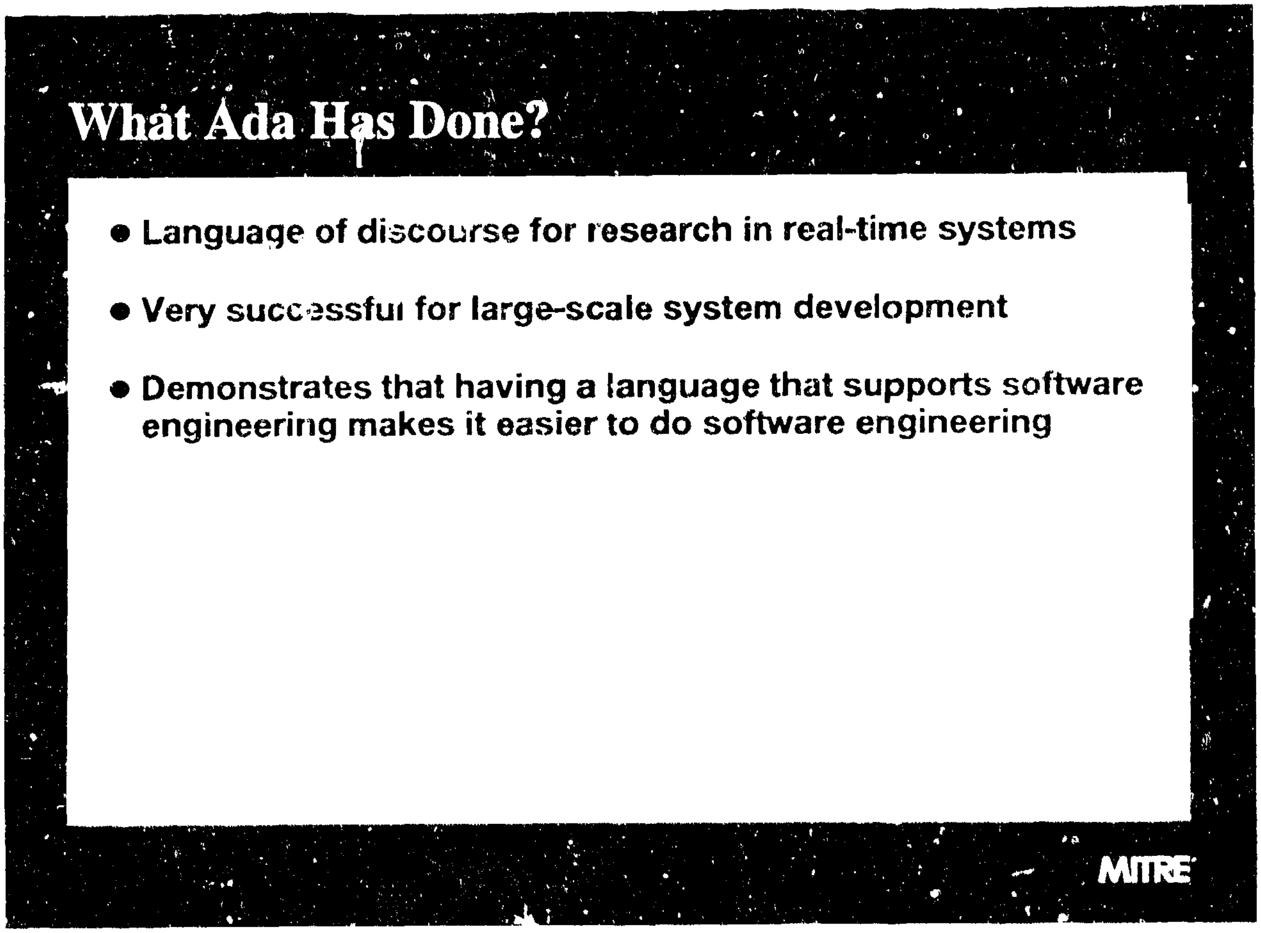
- DOD Language
  - Ada is perceived as a DOD language
  - High cost of validation and captive market causes high cost compiler problem
- Ada programmers more rounded
  - Ada programmers know several languages
  - Most C programmers only know C
    - UNIX and MS-DOS are the cause

MITRE



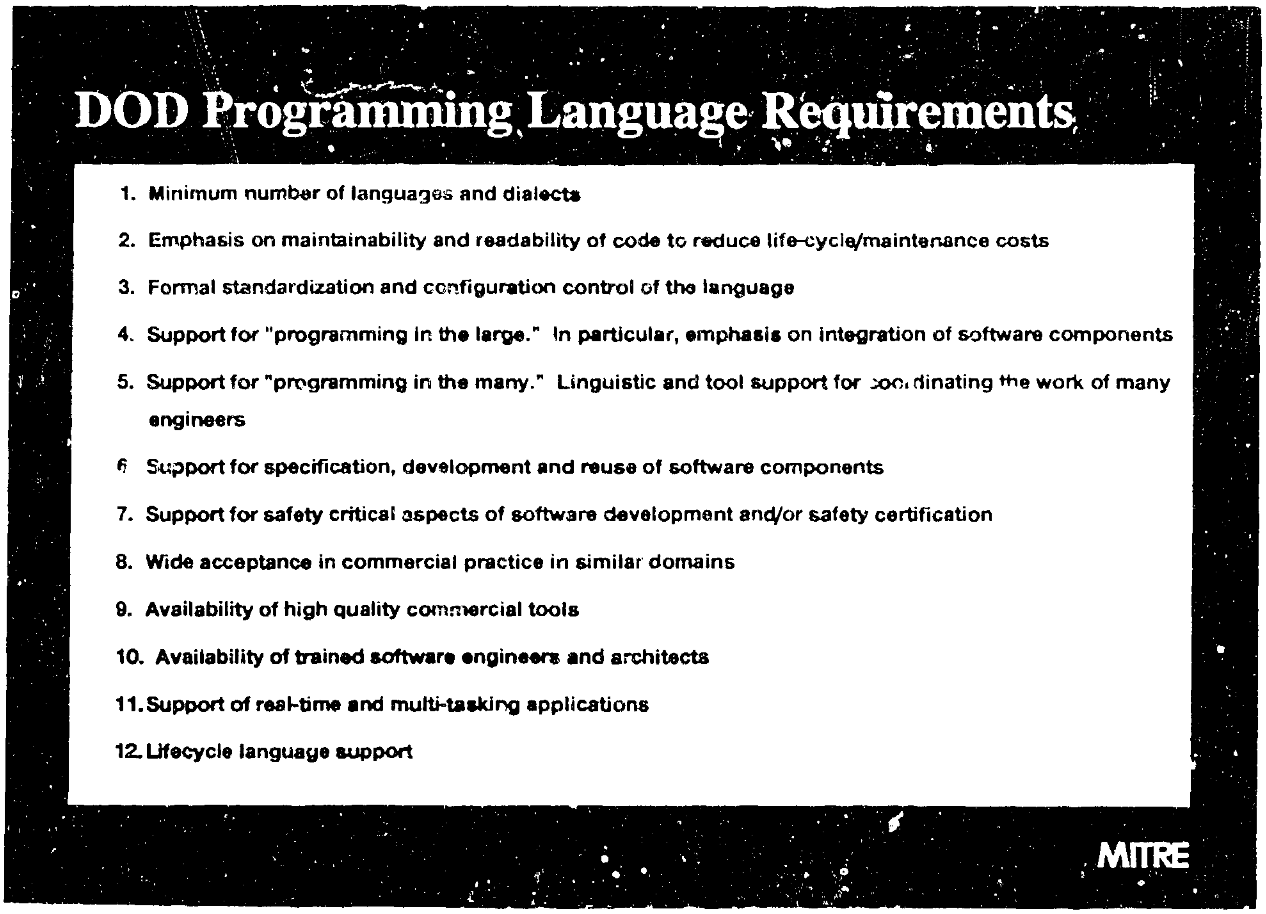
What Has Changed?

- Inexpensive compilers available
- Increasing interest in software engineering
- Number of languages in use is smaller
  - Most new code is done in COBOL, C/C++ and Ada in that order



What Ada Has Done?

- Language of discourse for research in real-time systems
- Very successful for large-scale system development
- Demonstrates that having a language that supports software engineering makes it easier to do software engineering



DOD Programming Language Requirements

1. Minimum number of languages and dialects
2. Emphasis on maintainability and readability of code to reduce life-cycle/maintenance costs
3. Formal standardization and configuration control of the language
4. Support for "programming in the large." In particular, emphasis on integration of software components
5. Support for "programming in the many." Linguistic and tool support for coordinating the work of many engineers
6 Support for specification, development and reuse of software components
7. Support for safety critical aspects of software development and/or safety certification
8. Wide acceptance in commercial practice in similar domains
9. Availability of high quality commercial tools
10. Availability of trained software engineers and architects
11. Support of real-time and multi-tasking applications
12. Lifecycle language support

MITRE

Participant Briefing
Gregory Aharonian
Source, Translation and Optimization

WEAK LINKS IN THE
COMMERCIALIZATION OF ADA

Gregory Aharonian
Source Translation & Optimization
P.O. Box 404, Belmont, MA 02178
617-489-3727 srctran@world.std.com

As a businessman considering investing more money to start new Ada businesses (or invest more in my Ada software reuse service), given the dual-use climate that is spreading, and DoD interest in commercializing Ada, I (and others) must look at the costs and benefits of doing so. Benefits are easy to recognize - more sales of Ada compilers, components and services. Given Ada's minor presence outside the Mandated world, I am sure some sort of linear relationship exists between the amount of money invested and the potential profits - the market is very elastic.

The costs of commercializing Ada is a much more complex subject, because of the existing behavior patterns for the supply and demand of programming products and programmers (i.e C/C++ is too entrenched and too competitive). In thinking about these costs, it is hard not to conclude that Ada is not very commercializable, short of sub-billion dollar investments by the Ada community. Other languages are too successful, and corporations making their next generation commitments now, for Ada to easily win market share. What follows is my proof on why Ada is not commercializable under current DoD economic assumptions.

The following argument is pessimistic for the sake of argument. I do not fully believe my own argument, as if I did, I would not be undertaking the great expense (at least for me) of attending the Ada Summit, nor would I continue to try and sustain my software reuse business in the Ada community. I make the argument so that those attending the Summit and readers on comp.lang.ada who have little contact with the non-Mandated business world can understand how difficult it is for those of us fighting for Ada acceptance. It also reflects what goes on in our minds as developments are announced inside and outside the Mandated world.

EFFORTS TO COMMERCIALIZE ADA WILL BE AS STRONG AS ITS WEAKEST LINKS.

WHAT FOLLOWS ARE THE WEAK LINKS.

---

1) The future standard platform has two Pentiums running Windows/Unix on which compilers cost $199 or less, a prohibitive price for Ada compiler vendors.

2) Jobs, jobs, jobs - there are 100 times more non-Ada jobs.

3) There is insufficient demand for an Ada components industry to bootstrap itself.

4) The embedded world does not like it's tasking built into the language.

5) It is too expensive to translate legacy projects in advanced software research from their existing languages to Ada.

6) The related worlds of signal processing, graphics, multimedia, networking and communications are dominated by companies closely tied to C/C++.

7) CASE tools make megaprogramming increasingly cost effective with C++ and Smalltalk.

8) Megaprogramming will increasingly become irrelevant.

9) The leaders in the academic world will never adopt Ada.

10) Growing apathy inside the Pentagon towards Ada

11) ARPA will never agree to put its enthusiasm and funding behind Ada.

12) The Ada Mandate will never force Ada compiler vendors to become competitive.

13) There is no marketing and demographic data for programming languages on which to make decisions and future plans.

14) Think of TCP/IP, CALS, VHDL, AHDL, SEI's CMM, all from the DoD. Their success implies Ada meets no pressing needs.

1) The future standard platform has two Pentiums running Windows/Unix on which compilers cost $199 or less, a prohibitive price for Ada compiler vendors.

The: four platform categories in which to market Ada: mainframes, minicomp workstations and microcomputers. The first three markets are too s to sustain a profitable business marketing Ada products.

Mainframes are a dying market - just look at the agonies of IBM and Fujitsu. Other than their massive communications and file serving ability, there is nothing attractive about mainframes. And even these two benefits are slowing being surpassed by other technologies. Even the DoD is getting rid of its mainframes for networks of small machines. The business that is still left in the mainframe market is mostly serving existing customers, and existing mainframe languages - Cobol and Fortran. Little demand there for Ada products, on top of competition from C vendors (such as SAS). And if nothing else, IBM has made it quite clear it is has no interest in promoting Ada in the mainframe world - kiss of death.

Minicomputers - same thing. Substitute Wang, Data General, DEC, Prime, Hewlett Packard and others (except for their workstation products) for IBM and Fujitsu and the same arguments hold. What business is left is servicing existing customers with their existing languages - little room for Ada products to sustain a profitable business. Again, while IBM has a strong market in minicomputers (ASXXX series), it has no interest in promoting Ada here also. (In fact, its upsetting that the leading STARS prime, IBM, refuses to commercialize Ada anywhere, preferring C++ and Smalltalk).

Workstations have three problems that limit Ada penetration in the market. First workstations were born out of the Unix/C world, and both C and C++ are heavily entrenched here. All of the big commercial software vendors, all the large university research efforts, and much of the communications and networking world using workstations are all based on C/C++. On top of which, many in the workstation world of the software hacker nature don't have a high regard for anything out of the DoD including Ada - a dumb but common attitude. Combine this with the Free Software Foundation, and the billions of lines of C/C++ workstation source code that float over the Internet every day, the many, many software research projects funded by the DoD in C++ et.al., and you have a tough, tough market to break into.

Second, there is not a big market for supporting products and libraries, to sustain sales of Ada products other than compilers and CASE tools. Can you think of any software product like 1-2-3 that drove the workstation market - a so-called "killer app" other than Unix itself? Given advertising and technical support costs, it would be hard to bootstrap a successful solely-Ada product into this market. And third, eventually all workstation software prices will coverge to PC prices (as PCs become more powerful and fast Windows and Macintosh emulators [WAPI for one] become available on workstations). Also GNAT will cannibalize some compiler sales in this market, and Cygnus-like companies can thrive but do not make an industry.

MICROCOMPUTERS

Which brings us to the microcomputer market, in particular, that market based on platforms with Intel microprocessors running Microsoft/Novell/IBM operating systems. As much as they may hope, all the non-Intel processor efforts (Sun, Motorola, MIPS) will never dislodge Intel from its dominant position. Intel is too good and too tough. Similarly no one will dislodge Microsoft, Novell, IBM and the Unix groups from their dominant operating system positions. It is here where the battle of Ada commercialization will be fought, a terrain where Ada is at a hugh disadvantage from the outset, having to charge up steep hills into the guns of C and C++.

For one, all of the dominant operating systems are C/C++ based: Windows, Novell OS, OS/2, and Unix. Microsoft alone has sold over 65,000 Windows NT developer kits at a cost comparable to that of Ada compilers. 65,000 is a number at least twice as great as all sales of Ada compilers, even including the Mandated world. These systems, and other applications, each total millions and millions of lines of C and C++. Most of the top 100 independent software vendors selling into this market use C or C++. Most of the new software technologies available for this market are done in C/C++: OO databases, multimedia, communications, multiprocessing, AI, natural language, etc.

New entrants usually succeed by having a small team code up a beta-product for version 1 to be marketed as quickly as possible. What's the language of choice for this SEI CMM-rating-of-0 development style? C/C++.

Second, there is intense competition between programming language vendors. Everyone is trying to get market share from Borland and Microsoft, who in turn seem to revel in fighting with each other. List prices for many compilers are often over $400, but if you present them with the cover sheet to a manual from a competitors products, they will chop off up to fifty percent of the price. Newcomers breaking in have to offer even lower prices than the market leaders, usually less than $200, and throw in some gimmick like a free CDROM or a trip to Hawaii. And whenever Borland feels a cash crunch, they make their prices $99. With their current cash streams, Ada compilers vendors would go bankrupt trying to compete in this market.

Third, the Ada community has ten years of non-relations with the press in the microcomputer market to make up for. There is little editorial coverage on Ada by the non-Mandated press (and even the Mandated press like GCN does as much to help Ada as hurt Ada with its coverage). There are movers and shakers in the microcomputer world, none of whom really are familiar with the Ada community. How can people decide to choose Ada if they never see it mentioned and reviewed in the publications they read?

Finally, not only does Ada have to battle heavily entrenched C/C++, but also other languages. Cobol is popular on PCs as the MIS world downsizes. Smalltalk is also quite popular, and winning some big converts, such as the Central Intelligence Agency, Xerox and American Airlines. Visual Basic from Microsoft will be quite popular (as its use as a Windows macro language). Lisp, Prolog, Forth, Pascal, and Objective C all have their little niches as big as the current Ada niche outside the Mandated world.

So if you break down the programming language market by platform, you will find that most platforms cannot provide sufficient sales to sustain a profitable Ada business, and that the one market that can sustain such businesses, the microcomputer market, is dominated by languages that are not only heavily entrenched but used to years of vicious competition.

[SOLUTION: Subsidize Ada compiler vendors to market the current versions of their PC compilers for $19.99, or some other really low amount. This is a tactic others in the PC business are using to quickly capture market share and profit off of future upgrades. Cost at least $15,000,000. Either that or nationalize Microsoft :-)

I constantly argue the government shouldn't do this stuff, but if ARPA is going to spend $100 million on commercial CAD, DoD can do so for Ada.]

2) Jobs, jobs, jobs - there are 100 times more non-Ada jobs.

Almost every economic commentary today revolves around jobs. NAFTA, Clinton's health plan, the Space Station, etc. Those who have them are in no mood to risk them, and those that want them will do what it takes to get the job. All of which is bad news for Ada programmers, because if you have been following the help wanted ads for programmers over the past few years, it becomes evident that there are over 100 times as many as non-Ada jobs outside the Mandated world as there are Ada jobs. 100 times. This is even more pronounced in the major software metropolitan areas like Route 128, Silicon Valley, Southern California, and Austin.

This sends out a very strong signal to the market to shift resources to those languages where there are new hires, because this is where many of the new programming language sales comes from. With growing dominance of microcomputers in the corporate computing world (see 1 above), there is the growing dominance of C/C++. And as the 1990 SEI infrastructure report mentions, and those familiar with terminai Markov chain states, once a programmer gets into the C/C++ market, there is little incentive to transition out. So as DoD downsizing continues, the job market will result in many qualified Ada programmers being absorbed by the C/C++ world. Nothing worse than a "Body Snatchers" scenario (and no, those are not pods being harvested on the remaining farms of the Silicon Valley and Boston areas - just funny shaped pumpkins and melons :-)

Thus the dynamics of the job market do not create many opportunities for new Ada sales. People sell where people are being hired and outfitted. Also see #9) below - academic world. Schools teach Ada because companies hire C/C++ programmers because schools teach Ada............

[SOLUTION: Fund a Works-Project-Administration-like effort by paying people to develop and initially market component libraries for Ada. See below. Cost $20,000,000.]

3) There is insufficient demand for an Ada components industry to bootstrap itself.

Markets are driven by people with a strong incentive to see their sector succeed. One reason why Apple's Macintosh became so popular, despite the higher prices of their machines, was that a lot of dedicated and profit hungry people wrote some great software for the Macintosh. In the programming language market, one of the great drivers is the supply of third party applications libraries: graphics, communications, databases, debuggers, memory management, GUIs, etc. A big factor in choosing a new language is the ability to find those libraries that you depend on in that library. For example, this explains one reason why the scientific groups never switched to Ada and slowly to C/C++ - no one has bothered to convert their massive Fortran numerical analysis libraries.

Many companies in the C/C++ world have grown from basements to nice sized companies selling these third party libraries and products, because in these markets people are willing to pay for reusable source code, even in object library format. More available libraries turns into more sales of compilers which results in more available libraries ad infinitum. A nice self-sustaining cycle that helps drive the C/C++ market, a process that is not passive. Also, the compiler companies work with third party suppliers and VARs to help this process along.

It is very difficult to start such a process for the Ada world. Given the marginal demand for Ada compilers outside the Mandated world, it is hard for a third party Ada component supplier to be profitable solely on sales outside the Mandated world, especially where there is so little support from Ada compiler vendors. Presently there are only a handful of Ada component suppliers, and the same ones everyone has been touting for the last ten years - Booch components, Grace components and a few others. (For some reason, CAMP seems to have dropped off the face of the earth). There is no growth, few successful new entrants - all indicators of a stagnating market.

One strategy might be to use sales inside the Mandated world to fund marketing and sales in the non-Mandated world, using the generous flows of money in the Mandated world to launch non-Mandated businesses. Nice idea in practice, but Mandated contractors are notorious for being unsupportive and disinterested in the third party component products, especially given often lamented but never changed 250+ DoD software procurement regulations, many of which hinder sales of reusable software to the DoD world (and in some cases make it illegal). Things are so bad, that it is illegal in the minds of many defense contractors, and probably in reality, to use charge numbers for contracts to allow workers to attend inhouse seminars to hear sales pitches for Ada products, though some companies are more open than others. (One reason current DoD reuse efforts are building on faulty economic models is that their workers never have had to actually sell reusable Ada code).

Another solution might be to lobby existing suppliers of C/C++ components to develop Ada versions. However since they have their hands full with maintaining their C/C++ versions, and would not recoup their investment in developing Ada versions, this will be a very hard sell.

In short, the climate is very chilly for those companies wanting to bootstrap third party Ada products and libraries, and without the drive that usually comes from these small companies, there are no vocal champions for commercializing Ada. The Ada compiler vendors have for ten years shown their disinterest in doing so.

[SOLUTION: Fund a Works-Project-Administration-like effort by paying people to develop and initially market component libraries for Ada. See # 2) solution. Subsidize C/C++ component suppliers to develop Ada versions. Cost $20,000,000]

4) The embedded world does not like it's tasking built into the language.

The growing power of embedded processors and their cheap costs make it possible to stick them in almost any product. This in turn fuels a booming embedded programming market, as evidenced by Embedded Systems Programming magazine and the Embedded Systems conferences. In following the embedded marketplace, I never get the feeling that programmers want tasking built into their language. There are dozens and dozens of multi- and real-time tasking executives for sale, each offering a different solution to the problem of many different instances of tasking in real-time products. Maybe they do want it in their compilers, but the don't seem to be flocking to Ada, and in the interim, these products help drive the C/C++ market.

Furthermore, as microcomputer operating systems become increasingly sophisticated (something like Windows NT), with tasking and other real time capabilities available as system calls, the demand for such capabilities explicitly in the language becomes a minor benefit,and sometimes a problem, as the programmer has to disable the programming language's tasking and real time features to use another executive.

And frankly, the Mandated world lost big time due to SEI's inability to work with CMU. Had they been able to get Ada used for developing Mach, which is being adopted by many as the operating system kernel, Ada would have benefited greatly. A tough sell (see workstation argument) but at the time it would have been worth the effort.

Thus outside the Mandated world, I am not sure how much benefit it is for Ada to have tasking built into it. Inside the Mandated world, sure it is of great to help for contributing to the reliability and maintainability of embedded weapon systems. But outside, I am not sure if it is much of a advantage for marketing and commercialization. I have heard good stories from the Tartan people about their non-Mandated sales, but this might be all of the demand.

[SOLUTION: This is a tough one, since it is an issue of personal preference on the part of embedded systems designers. A first step would be to do an indepth survey of the non-Mandated embedded systems community to find out what they are doing, how they are doing it, and why they are doing it. Then maybe some angle can be found to sell Ada. Cost of study: $200,000]

5) It is too expensive to translate legacy projects in advanced software research from their existing languages to Ada.

Programming language sales come from two places - new efforts and existing efforts. Existing efforts usually mean legacy projects, and in many programming communities, this means large amounts of code that have to be converted before a new language can be adopted. In the scientific programming world, no one will switch unless their large numerical analysis libraries are converted. In the workstation and academic worlds (which until quite recently was the same), you are talking about legacy projects in C and C++. In all of these worlds, you are talking about hundreds of millions of lines of code in an existing language.

You may argue bindings and Ada pragmas, but this has never been a popular approach in any language. Besides even here, Ada is being beaten out by other languages. For example, ParcPlace has a nice system for practically wrapping Cobol modules inside of Smalltalk objects, so that gradually a top-down reengineering of Cobol legacy systems can be done in a manageable and cost effective way. I am not arguing that this is a good or bad thing to do, but that for those doing it, other languages are beating Ada to the marketplace. OMG efforts with CORBA also make other languages equally attractive for OO as compared to Ada9X.

Even in the DoD world, there are popular legacy systems that have never been converted to Ada as part of some evangelizing effort. LOWTRAN and ASCA from the Air Force; GRASS, BLAST and BRLCAD from the Army; NPSNET and NEC from the Navy and other programs are popular, important and still not in Ada. Many of the research projects at the service academies qualify as legacy projects, and most are not being done in Ada, which is a shame, since the service academies (especially the NPS and AFIT) do some great work. And it would have been nice if someone had funded a Jovial-to-Ada converter years ago. Every four months I get a swarm of phonecalls as some Jovial conversion effort hits the street and everyone starts looking for a Jovial converter, yet no one individually has the funding or interest to have one developed (yea, I know translators never work-look at the millions spent on the now forgotten WWMMCS Fortran and Cobol translators), but a decent set of assistance tools could be developed.

Thus for those interested in pursuing commercialization through thrusts into the academic and corporate worlds,there is a big problem of converting legacy systems into Ada before or in parallel with the adoption of Ada. It is a very tough sell, and risky investment for all parties.

[SOLUTION: Fund the translation of DoD and non-DoD popular legacy systems. Why setled for just GNAT - why not Ada versions of all of the FSF tools? Help some of the large university legacy systems to convert to Ada (for example, SPICE. Cost at least $20 million, preceded by a $100,000 study to do a census to find out where the legacy systems are].

6) The related worlds of signal processing, graphics, multimedia, networking and communications are dominated by companies closely tied to C/C++.

There is a lot of activity, funding, press, excitement in the growing worlds of global networking, multimedia, entertainment and virtual reality. All of the entertainment companies, the telephone and cable companies, and many software developers are jumping into this market. It is estimated that everyday in the San Francisco Bay Area one new multimedia company is being formed. Tens of billons of dollars are being thrown around with reckless abandon (not really - these people are bottom-liners all the way), with Japan and its billions also joining in.

Unfortunately for commercializing Ada, there is little Ada use by these collaborating groups. The entertainment companies don't program, the cable companies follow telecommunications companies, the telecommunication companies (AT&T, Baby Bells, Sprint, MCI) are all using C/C++, and the software companies in multimedia and graphics, with their Macintoshes and PCs, are also all using C/C++. Sure there is the occasional company like Silicon Graphics successfully and proudly using Ada to do this type of work, but economically they have only a marginal presence.

If you read the magazines for electrical engineers (Electronic Design, Electronic Engineering Times, Embedded Systems Programming) where the hardware for the global multimedia world is covered, all you see is ads for C/C++ products and tools. Ditto for the software developers (Computer Graphics, New Age Media, Wired, many of the ACM and IEEE magazines). Global interactive multimedia communications is the next big industry - and Ada has no presence at its birth.

And someone stop the Navy and Air Force from showing their really nice distributed graphics programming system (NPSNet) written in C++ to the graphics world (also funded by ARPA - C++ being distributed to industry). I know that they are proud, and they should be for their effort, but it is kind of embarassing for the Mandated World that it is not in Ada.

Again, commercializing Ada in this marketplace will be quite hard, for the usual reasons of entrenched first-entry-ers. (I could go on with similar stories for the AI, parallel and multiprocssing, database and other domains, but it is just the same as here). Too much money is being spent in the multimedia world right now, locking in people to programming language choices in the near future, or some later day Ada marketing campaign to have much of an impact.

[SOLUTION: Bribe Steven Speilberg, or fund Silicon Graphics to develop some nice multimedia tools in Ada, and donate lots of their platforms into the multimedia industry. Cost at least $10,000,000. For other areas of software application, initially target the advanced software R&D in those areas (parallel, OO, AI, etc), which conveniently has a lot funded by ARPA - see #11) below]

7) CASE tools make megaprogramming increasingly cost effective with C++ and Smalltalk.

Any campaign to commercialize Ada will have to be founded on whatever strengths Ada does have in relation to other languages. Once such strength is the use of the language in megaprogramming projects, those large team multi-programming projects prevalent in the Defense world. For these projects, use of Ada has consistently shown the benefit of doing so. (Of course it would help if some of the megaprojects [FAA, WWMCCS, STANFINS] didn't get into so much trouble and bad press for non-Ada reasons).

The problem in commercializing Ada using the megaprogamming benefit as a selling point is that increasingly, CASE tools are making megaprogramming possible with other languages. For example, at a 1991 OOPSLA conference, someone from Rational argued this very point - that Rational's tools can be used to allow development of > 100,000 lines of code projects in C++. If you attend commercial CASE shows, you see more and more tools from other companies offering similar capabilities. And success stories from companies using C++ for large projects (ATT, Microsoft, Mentor) and Smalltalk (Xerox, American Airlines) only strengthens the case for these languages.

I am tired of Ada people bragging about the strong type checking you get with Ada. This is a lousy marketing argument, since such features are increasingly available with CASE tools for other languages. I have heard this brag being made for Ada9X. Well it is no longer an impressive feature.

Now it may turn out that in fact using languages other than Ada for megaprogramming is not as cost effective as some may claim, that there are hidden costs that have been ignored or not accounted for. But from a marketing point of view, enough of a case for non-Ada megaprogramming can be made to weaken this approach to commercializing Ada.

[SOLUTION: Again, another tough one. Given the "equivalent" capabilities of Ada9X and C++ and Smalltalk, it is hard to design a CASE tool for Ada that can't equally as well be used for these other languages The only hope here is to loudly and endlessly publicize success stories with Ada projects to at least keep Ada's good news in the public's eye. Cost is $2,000,000 to fund a professional publicity campaign. And please, can someone at gunpoint force the STARS people to start showing up at the commercial trade shows and conferences dealing with CASE?]

8) Megaprogramming will increasingly become irrelevant.

Of course, this even assumes that megaprogramming should be done at all. Most of the trends in the future computing marketplace, the world of networked Intel microprocessors, is to have general shells, both for operating system and applications, in which sharable, reusable modules are developed. The DLL technology Microsoft has championed is one approach, while Lotus NOTES system is an example of the general shell environment in which smaller subsystems do their thing. For example, the Air Force has had some success in using C++ based groupware for building large MIS-like systems (though for Ada's sake they didn't have to brag about it in Government Computer News).

What few megaprogramming projects there are, the shells, will most likely be legacy systems already in existence or hatching now, and they will all be in C/C++. Windows, Solaris, Motif, Notes, groupware, etc are all competing to be dominant in this area. Sure in some industries (avionics always being touted) megaprogramming will have some role, with wins for Ada, but never enough to sustain a thriving Ada industry.

As the corporations downsize both themselves and their information processing activities, mainframes and megaprogramming will gradually diminish in scope, and while providing sales to an installed base, will present few opportunities to sustain healthy Ada sales. And it certainly wouldn't hurt for the DoD to question its reliance on megaprogramming - the wars of the future might be better served with distributed systems - it works for the wars of commerce.

[SOLUTION: There is none here. The future is not megaprogramming. If you want to know the future, read the science fiction book "Ender's Game" by Orson Scott Card, to see how the large distributed simulation wargames the DoD is increasingly using (all in C/C++) will eventually blur the line between virtual and live combat.]

9) The leaders in the academic world will never adopt Ada.

This follows from a lot of the other sections, but I mention it explicitly becuase so many in the Ada community seem to be placing their hopes on using GNAT in the academic community to launch some campaign to eventually win market share for Ada. I am dubious, for the following reasons.

First academic success of a language means little. For example, the most popular undergraduate teaching language is Pascal, but it is rarely used in the corporate world (compared to Cobol/C/C++/Fortran). And if an academic institution geared its language choice vocationally, it would choose C/C++ as the professors read the help wanted ads, and hear from their former students looking to hire part-time people. An argument can be made for learning with the better language, Ada, but in tight job markets, people want what is being hired on their resumes.

Second, most retained academic computing experience comes from work on academic research projects as a junior or senior, or graduate student. However, most of these legacy projects are not in Ada, and will rarely if ever be translated. So even though you might expose a student to Ada as a freshman or sophmore in the introductory courses, most of the advanced courses and work will not be done in Ada, though will be the most recent experiences to retain.

Third, NOT TO DENIGRATE THEIR CONTRIBUTIONS OR RELEVANCE TO SOCIETY, but most of the institutions teaching Ada or where Ada can be introduced, are not the "leading" software academic institutions. Most of the top software institutions, as measured by the presence of their students in those companies dominating, driving and creating new markets, by their presence in leading software metropolitan areas, by their receipt of the advanced software research government funding from the DoD, DoE, NASA, NSF and NIST, for the most part don't use or care for Ada. MIT, Harvard, Carnegie Mellon, Stanford, Berkeley and the rest of the UCAL system, Cornell, etc, are not and probably never will be Ada institutions.

Fourth, the service academies don't have the best record for using Ada. If you measure the programming language use in theses issued by the NPS and AFIT, you will see Fortran, Pascal, C/C++ and Ada equally distributed - better than academic institutions in general, but not the most toutable record for DoD service academies. It's a shame, because the work I see in these theses is first rate and would be a big credit to Ada if it was all done in Ada.

Thus, I am dubious about the strategy of commercializing Ada through the route of academic acceptance, especially if Microsoft starts flooding the universities with the source code to Windows NT. GNAT helps, but only as a skirmish in a much larger battle.

[SOLUTION: As part of #2), #3) and #5), fund efforts in the major universites to convert some of their legacy projects into Ada. Also flood the service academies with Ada compilers. Make ARPA tie academic funding to Ada use.]

10) Growing apathy and bickering inside the Pentagon towards Ada

To quote Ralph Crafts, "there is a resentment and fear in the various components of the DoD regarding the Ada directive", that he has been told by lower ranking Pentagon officials that "no one has ever explained to them why Ada is a good idea or how Ada can help them in accomplishing their mission"; instead they have been told "thou shalt use Ada" without a shred of accompanying justification and rationale.

My estimate, based on how people inside the DoD spends their money, and not on what people say, is that the majority of the DoD is either apathetic or hostile to Ada. My estimate is as follows - at least 95% of ARPA is apathetic towards Ada (see below), at least 75% of the Navy (based on Tuttle's and other Navy comments, and comments from Navy contractors), while the Air Force and Army are probably split down the middle. For example, the Air Force funds the KBSA project, a CASE effort with similar goals to STARS, but KBSA is not being done with Ada, and the Air Force is so happy with KBSA results it is funding commercializing KBSA technology, offsetting good Air Force things done with Ada. Another example, in an Army study of Army expert systems, 19 Army expert systems were reviewed, none written in Ada (most in Lisp). The service academies have shown lukewarm support for Ada. DISA is too new to guess about, and while holding the Ada summit, DISA's recent decision to buy lots of Windows NT systems basically helps C++. The exact percentage? I don't know and I know that no one else inside the DoD knows, including DISA.

Collectively these numbers add up to greater than 50% disinterest in Ada inside the DoD proper. What do I base this on? Admittedly my data is probably not deep enough statistically (see #13 below), but I do get around. For example, there have been a decreasing number of DoD SBIR topics involving Ada, with the most recent 1994 solicitation completely devoid of any mentions of Ada. The majority of the success articles that I read for Defense programming activities use non-Ada languages. The majority of source code listed in DoD service academy theses and technical reports, and general DoD technical reports, is not Ada. Sure the large, mega projects inside the DoD are in Ada, and should be, but the majority of everything else is not in Ada.

As many have noted publicly and privately (I have endless quotes if you want them), the Ada waiver process is a joke, which wouldn't be if the DoD was truly behind Ada. In recent months, a variety of DoD officials (for example, General Kind) have suggested that maybe Ada needs some reexamination. To quote Ralph Crafts, "There's been no continuity, enforcement or explanation to the people in the field who are supposed to implement the Ada requirement. I've been with several organizations over the last few months, and there's confusion, resentment and anger because they get the word one way one month and then some high-level person comes in and says 'Just go ahead and do it this way. Ignore the Mandate until we get caught." and "The Ada waiver is ludicrous. There are claims by the services that they are fully behind Ada, and they have granted no waivers or a handful of waivers over the last couple of years. Well, that's a really nice facade because people who don't want to use Ada don't bother to go through the waiver process. They go off and do what they want and hope they don't get caught until they have so much invested in it that they can't be told to stop. It's interesting that people can just ignore the Ada Mandate with impunity and they will go off and require the use of C and C++".

And then there are ancedotes like the following: "I watched in amusement (for five years, no less) senior officers in the Air Force decide that their old COBOL systems needed only 30% of their software rewritten to repair the junk. The only reason they stopped at 30 was because they had to use Ada if it hit 33%. Of course, two years later, they'd REestimate that ANOTHER 30% had to be rewritten -- deftly avoiding the Mandate without fear of getting into trouble."

And it doesn't help when DoD officials like Paul Strassman, once they are out of office, trash Ada publicly. Many in the corporate world read of these bickerings in the trade press, and use it as an excuse to not consider Ada. To quote Paul Strassman, "The economic driver really is software reuse. The underlying reason for Ada originally was software reuse and software modularity. Today, by and large, software reuse is being driven by the fact that software objects are commercially available, and most of the software objects on the market are just no Ada objects. The world is not going the Ada way". He doesn't mention that because DoD contractors don't like buying Ada objects, most of the few Ada components suppliers can't remain in business. But in any event, his comments, as an ex-DoD software official, hurts Ada credibility.

So in general, the unity of attitude towards Ada that some in the DoD claims exists, doesn't exist. There is a lot of conflicting attitudes, concerns, frustrations, and programming language practices.

[SOLUTION: I have absolutely nothing to suggest here. Had I been able to figure out this messy politics, I would have figured out how to win a contract. In six years I haven't done the latter, so don't expect much from me on the former.]

11) ARPA will never agree to put its enthusiasm and funding behind Ada.

An anonymous ARPA Ada quote, seen in 9/14/92, pg 98 Government Computer News, "It makes absolutely no sense to have a law to mandate the use of Ada. It was not the intent of the Ada program to have absolutely one programming languge. The law CANNOT apply to ARPA."

In the eyes of many decision makers in the corporate world, ARPA is the DoD's leading software agency. ARPA funds all of the advanced research in software and has a reputation for starting computing sub-industries. And when ARPA likes something, it goes after it enthusiastically, to the point of crossing over political policy lines. Craig Fields was willing to be fired as director of ARPA when he tried to inject some venture funding in a semiconductor company, crucial apparently, to help ARPA needs. GAO and Congress chastised ARPA for its meddling in the supercomputing market, where it directed companies to purchase and use certain machines favorite to ARPA. Recently ARPA anted up $100 million to help commercialize the CAD industry vis-a-vis DoD needs, with Martin Marietta receiving $53.4 million and Lockheed Sanders $42.5 million. And then there are the hundreds of millions that have gone into SEMATECH. ARPA will also fund up to $48 million to help the active matrix liquid crystal display industry. So when ARPA likes something, it REALLY likes something.

Here's an example of a project ARPA could have forced Ada to be used, but didn't, and an example of how they spend big bucks on everything but Ada. ARPA is funding the Center for High Performance Computing (CHPC) with $15 million to create a real-time operating system scheduling framework, distributed file systems, a high-performance fiber-optic connection and multi-processor modules to work with the OSF's Mach 3 microkernel (again where was SEI with CMU?). Ada is NOT being used by CHPC to do this work. To quote CPHC director (an ex-Marine), "I'm a big Ada advocate, but commercial developers just aren't producing the kind of Ada products that would make it catch on. The realities are such that you aren't going to see commercial software done in Ada. When companies like Microsoft make Ada compilers, then I'll believe Ada has a future". Why should commercial developers use Ada when ARPA refuses to make this project, and others, use Ada?

Yet when it doesn't like something, it is apathetic to hostile, like with Ada. More than a few ARPA officials have been quoted, for retribution and anonymously, that the Ada Mandate does not apply to ARPA. ARPA funding of advanced software research in non-Ada languages is at least ten times larger than its Ada funding. And since the commercial CAD industry is as important to the DoD as is the commercial Ada industry, and a lot more healthy than the commercial Ada industry, certainly whatever reasoning led to aiding the commercial CAD industry with $100 million should have led ARPA to a similar decision for Ada.

And the Ada ARPA does get involved with, like STARS, seems more like fifth column undertakings than anything else (and until someone from STARS shows up at a commercial CASE show and starts displaying the benefits so often touted within the STARS community, few outside the Mandated world are going to take STARS seriously; if the Air Force can commercialize the non-Ada XBSA, certainly ARPA can try commercializing STARS).

Commercializing Ada requires handling perceptions. And one perception to many in the corporate world (or at least those who bother to follow Ada at all) is that the DoD's leading software agency does not like Ada. An agency that many in the corporate world pay attention to, if for nothing else than the fact that ARPA doesn't get reorganized every few years.

[SOLUTION: Quite simply, does DISA have the blood and guts to fight ARPA, and have the DoD force ARPA to embrace Ada? If not, then this whole thing is a complete waste of time. If we can't sell ARPA on Ada, we won't be able to sell anyone else on Ada. For those of us spending our own money on trying to commercialize Ada, it is frustrating seeing the DoD continualy ducking this issue. DISA needs to put ARPA into a headlock over Ada. Maybe give a contract to Hulk Hogan.]

12) The Ada Mandate will never force Ada compiler vendors to become competitive.

In a world where the DoD disconnects itself from the general software industry, and is willing to pay any cost to develop defense software systems, the Ada Mandate is a viable, beneficial mandate to have. Use one language - simple, clear and very conducive to reuse and maintenance.

In any other world, the Ada Mandate is a failed microeconomic distortion of the marketplace. At their current price levels and capabilities, Ada compilers are uncompetitive. To compete in the microcomputer marketplace, and even the workstation marketplace, Ada vendors will only be able to win market share by charging $99 or less for their compilers. Yet once they do so, they have to charge the DoD and its contractors the same amount, and as they have admitted, they cannot afford to do so. As long as the Ada Mandate is in its current form, the DoD can have one language or it can have competitive Ada compliers, but not both.

[To show how vicious PC pricing is, Borland is trying to capture market share, and pass Lotus to be number two in the PC spreadsheet market, by selling the latest version of their Quattro spreadsheet for $49, 90% off the list price of $495. Any Ada compiler vendors want to offer 90% discounts to everyone to steal some market share from C/C++?]

Further, Ada credibility is weakened by Ada compiler vendors trying to survive at any costs. I have seen more ads from Ada vendors and contractors marketing C++ products than I have seen for Ada products. I can understand the Ada vendors trying to survive, but it is at the expense of their own Ada sales, and testimony to the problems caused by the market distortions of the Ada Mandate. In recent months, I have seen ads from Telesoft for their TeleUse C++ GUI code generator, Intermetrics for their 68000 C compilers for the embedded world, many from IBM pushing C++ and Smalltalk, Meridian with their C++ Designer, DDC-I with their C++ multitasking executive, Rational selling the Booch components in C++ with their Rational Rose, and Tartan selling a C++ DSP compiler. In aggregrate, as much is being spent by Ada vendors on these non-Ada product advertisements as for their Ada products if not more - understandable but not credible.

[SOLUTION: Unfortunately, while I can see ways to address many of the other problems cited above, fixing the Mandate escapes me. Get rid of it, and the vendors can't survive on their Ada sales. Expand it to include C++, and the same thing happers. Even worse, the longer this non-solvable issue goes unsolved, the more costly it will be for the DoD to solve it.

One suggestion would be to subsidize the Ada compiler vendors to flood the markets with $19.99 pricing on the current versions of their compilers, in order to profit in the future from upgrades to a much larger installed base. But frankly, I don't think they should be awarded in this way, given their past histories. After all, they should have be the ones organizing this Ada Summit, not DISA. Though if nothing else for their advertising consistency over the past few years, I could see funding Alsys and Aetech.]

13) There is no marketing and demographic data for programming languages on which to make decisions and future plans.

One vital element in any business plan, especially when you are approaching venture capitalists, is the market survey. This involves determining the potential demand for your product, and current suppliers that you will be competing with. Such data does not exist for the programming language market vis-a-vis Ada interests. I have heard that DISA is secretly developing plans to "save" Ada - well if they are (why in secret?), they should stop if they don't have the following data.

In any event, inside and outside the Mandated World, we need to know: what programming languages are being used, who is using them, for what applications are these languages being used, trends in pricing for programming language products, which magazines and advertising channels decision makers use, schedules and popularity of commercial software engineering trade shows and conferences, dynamics of software development at the top 200 independent software vendors, academic use of programming languages for both teaching and R&D, DoD use of programming languages at all scales and for all applications, flows or reusable software, and other marketing factors.

Such data is vital for successfully commercializing Ada, and it is collectable, as I have demonstrated with my occasional statistics postings to comp.lang.ada. To proceed without such data is a very risky gamble that most likely will fail.

[SOLUTION: Fund such a study at a cost of $500,000.]

14) Think of TCP/IP, CALS, VHDL, AHDL, SEI's CMM, all from the DoD. Their success implies Ada meets no pressing needs.

Surprisingly for a government bureacracy, the DoD has had some successes in developing software technologies that the general software industry accepted commercially and profitably for entrepreneurs. TCP/IP, CALS, VHDL, AHDL (the new analog design language), and SEI's CMM rating have all enjoyed commercial success. Why? Each met some need industry had, and each were a good solution for that need, all of which have spawned many self sustaining companies and/or incorporated intc product lines. Even for these successes, it took many years of use to gain wide acceptance. Overnight stardom could not be mandated.

Ada has not had the same degree of acceptance and success. It is not too rash then to conclude it mets no pressing need, and probably won't fill any pressing need for the next few years. Thus commercialization becomes difficult to achieve.

[SOLUTION: Look 5 out of 6 success with software technologies is not a bad record, more successful than my ventures. Chalk up the defeat to learning, and move on].

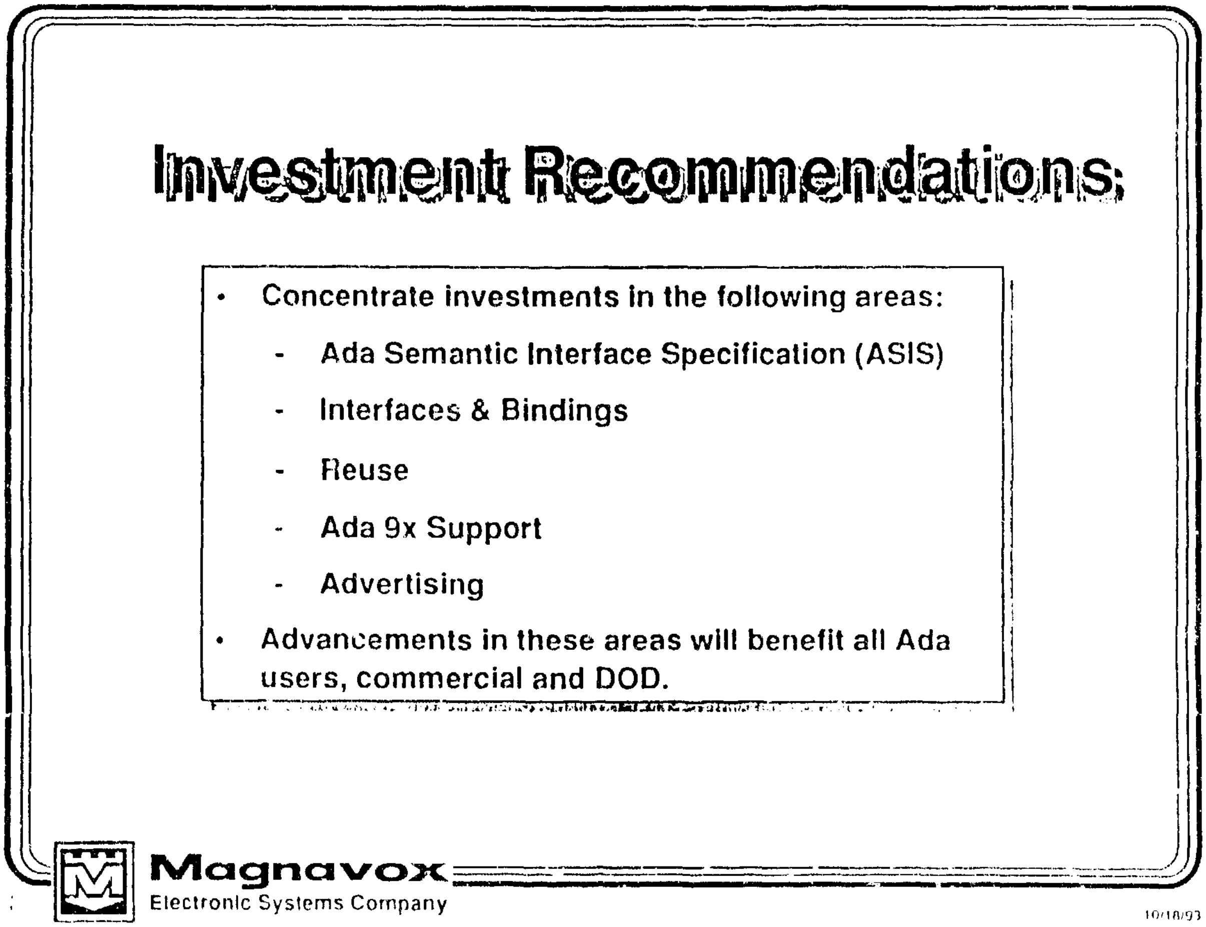
Participant Briefing
Scott Stevens
Magnavox

Ada Dual-Use Workshop

Presentation provided

by

Magnavox
Electronic Systems Company



Investment Recommendations:

- Concentrate investments in the following areas:
  - Ada Semantic Interface Specification (ASIS)
  - Interfaces & Bindings
  - Reuse
  - Ada 9x Support
  - Advertising
- Advancements in these areas will benefit all Ada users, commercial and DOD.

Magnavox
Electronic Systems Company

10/18/93

ASIS

- These interfaces will allow tool builders to access detailed semantic information, without investing in a compiler front-end.
- With these interfaces, a whole new generation of support tools can be developed.
- Support for AS!S version 1.1 (Ada 83) is slowly building, but could use a boost.
- Version 2.0 (Ada 9x) should be expedited, so that 9x can be supported in this area.

Interfaces & Bindings

- Quality Ada interfaces to commercial off-the-self (COTS) products are badly needed.
- These interfaces need to support multiple integration strategies.
- To be competitive with environments which do not require bindings, they need to be readily available and very inexpensive.

Reuse

- Continue efforts to make large-scale reuse libraries available to both defense and commercial users.
- To be cost effective, developers must be able to quickly locate and retrieve high quality components.
- Incentives must be provided to both DOD and commercial developers to provide quality components to the libraries.

Ada 9x

- These improvements will make Ada a serious contender in the object-oriented programming world.
- New language features will help address many of the issues which have hindered the acceptance of Ada.
- Venders need to be "encouraged" to provide timely support for Ada 9x, including the annexes.

Advertise

- It is time to begin marketing the Ada alternative, outside of the defense community.
- Almost ten years of experience with Ada in the defense community should be shared with the commercial sector.
- From a technical perspective, Ada 9x will compare favorably to the alternatives.
- From a business standpoint, it will take effective reuse, and superior programming support environments to win the business.

Summary

- All of the issues presented are being reviewed at some level by the Government or industry groups.
- The committees, groups, action teams and agencies need adequate funding to finish their tasks in a timely manner.
- In addition to these groups, commercial vendors will need "incentives" to help offset the costs associated with providing support for ASIS and Ada 9x within a relatively small Ada marketplace .

Ada Dual-Use Workshop

Recommendations Provided By

Magnavox Electronic Systems Company

Introduction

Magnavox has been using Ada on large Command and Control applications since 1984. In fact, we have actively tracked and been involved with the language that became Ada since the inception of the Higher-Order Language Working Group (HOLWG) in the mid-1970's.

It is from this inside perspective that we offer an integrated response to the questions posed by this panel:

- What actions can we take to get Ada accepted as the preferred language for development of software systems within your organization?
- What investments can we make to increase the commercial use of Ada?
- What can we do to increase Ada's appeal, adoption and popularity within your software community?

Response to Questions

As a software developer for the DOD, Ada is the required and preferred language for most of our large projects. In addition to these large systems, we also produce many small embedded systems in which C is currently the language of choice for various reasons. In order for Ada to become the preferred language company-wide, the following must be true:

- Ada compilers must be able to operate with a minimum of resources. Many smaller projects are developed with only a few PCs or workstations as the programming environment--these cannot provide the resources needed by many of the Ada development environments.
- Compiled code must be extremely optimized in both size and performance. In some cases we have switched to Ada because improvements in computer and compiler technology have made the use of Ada feasible for the first time. Some of our embedded system products are extremely small with stringent size, power, weight and performance constraints. As technology has improved we have been able to use Ada in applications that previously required carefully crafted assembly language.
- Ada programming environments must be affordable. Small projects can not easily afford the high cost of today's Ada environment, which is mainly targeted to large-scale development.

There are many issues related to the commercial acceptance of Ada, many of which will most likely be covered by participants from others areas. We would like to see investments in the following areas:

ASIS

The recent introduction of the ASIS (Ada Sematic Interface Specification) provides the opportunity for Ada environments to become far superior to other languages in terms of tool integration. Magnavox has successfully used the LRM interfaces (a precursor to ASIS) provided in the Rational environment, to build several tools for automatic code generation. These tools could not have been produced in a cost effective manner, without these interfaces.

Ada compiler and tool vendors must be given the proper encouragement to quickly adopt this standard and produce integrated development environments. The effort to produce an ASIS interface for Ada 9x should also be expedited.

Interfaces & Binding

Both the commercial and defense worlds are struggling to integrate available components into large systems which meet customer and market needs. The challenges of large scale systems integration in an Ada environment is further complicated by the lack of Ada interfaces to many of these commercial products. Even though the Ada language provides the capability for a developer to integrate multiple languages into a single program, the practical aspects of maintaining a development environment for more than one language can greatly increase developmental costs.

Quality Ada interfaces to commercial off-the-self products (COTS) are needed to provide a more cost effective environment for Ada-based integration of commercial and defense-related systems. These interfaces need to offer multiple integration strategies that are flexible and allow the developer some implementation options. Interface paradigms such as client-server, blocking/non-blocking, and type conversion issues need to be addressed by these interfaces, to allow a seamless integration with the Ada environment.

Developers need C and C++ binding to products such as windowing environments, databases, and operating system services. Significant effort has been focused on this area, but reliable interfaces are not easy to locate. These bindings must be readily available for the many different products, and must be free or extremely inexpensive, in order for Ada to compete with environments which do not require bindings.

Reuse

This is, of course, one of the hottest topics in the software development community. The Government needs to continue the efforts to make large scale reuse libraries available to both defense and commercial users. Cost effective use of these libraries depends on the ease in which developers can quickly locate and retrieve desired components. To gain confidence in these libraries, the components must be highly reliable and well maintained. Incentives must be provided to both DOD and commercial developers to provide quality components to keep the libraries well stocked.

Ada 9x

Ada 9x represents a significant and long awaited improvement to the Ada programming language. These improvements will allow Ada to become a serious contender in the object-oriented programming world. The new features provided by Ada 9x will allow Ada to be used for many of our small embedded systems. The question is, who will support Ada 9x in terms of compilers, language sensitive editors, debugging facilities and other programming support tools? Proper incentives should be provided to vendors to help get Ada 9x off to a running start. Vendors should be encouraged to provide full support for the language including the annexes.

Advertising

The marketing "word" on how to cost effectively implement Ada and obtain benefits (technical, financial, time-to-market, quality, etc.) must get out through traditional industry publications such as ComputerWorld, InfoWord, PCWeek, etc. It is time to share the experiences of the defense community with the commercial world so that they can get a boost from almost ten years of experience with Ada.

Summary

The question asked was "what can we [Government] do...?" Most of the areas addressed in this paper are being reviewed at some level by the Government. The committees, working groups, Action Teams, and agencies need receive proper funding to complete their tasks in a timely manner.

In addition to these groups, commercial vendors will need "incentives" to help offset the costs associated with providing support for ASIS and Ada 9x within the relatively small Ada marketplace.

Continued support in both of these areas will be needed until Ada gains enough of the market share to become industry supported.

Participant Briefing
Bruce Krell
Hughes

Ada Summit

Increasing Commercial Use Of Ada

Dr. Bruce Krell
October 19-20, 1993

MY CREDENTIALS

DoD Project	TOW Missile Control For Cobra Helicopters
Commercial Project	Electrophysiology Analysis System For Cardiologists
DoD	Developing With Ada: Life Cycle Methods by Dr. Bruce Krell, Bantam Professional Books, 1992
Commercial	High Speed Windows Applications: Multitasking Design Methods by Dr. Bruce Krell, Bantam Professional Books, 1993
Multilingual	Ada, C, C++, Fortran, Cobol, Pascal, DBASE, SQL

Increasing Preference For Ada In Aerospace/Defense

HUGHES

Short Term

Uniformly, Rigorously , Enforce Ada Mandate:

- Deliverable Systems
- Military Research Labs

Avoid Herd Mentality Approach To Development:

- CASE
- Reuse
- Object-Oriented

Abandon OOP Aspects Of Ada 9X:

- Inheritance
- Run-Time Polymorphism

Long Term

Educate DoD Program Managers And Staff:

- Importance Of SW Engineering
- Effective Tailoring Of 2167A

Require Certification Of Software Engineers:

- Standard Training: Initial, Continuous
- Certification Examination
- Design Approval Authority Before Coding

Allow Percentage Of Procurement Funding

- For Training In SW Engineering, Not Coding

Increasing Preference For Ada Within Hughes

HUGHES

General Experiences

Assuming: Rigorous software engineering prior to coding
Effective usage of Ada Characteristics

Observed:		
	Requirements/Design	55% Of Schedule
	Code/Unit Test:	20% Of Schedule
	Integration Test:	25% Of Schedule

Better Performance
Higher Reliability
No Integration And Test Nightmares
Delivery Within Cost/Schedule Constraints

Controlled Experimentation

Same Design Implemented In Both Ada, C
Design Uses Encapsulation (objects. ADTs)
Fixed Development Periods (15 weeks)

100% of Ada Users Delivered 100% of Integrated Code
70% of C Users Delivered 70% of Integrated Code

Increasing Commercial Use Of Ada-The Problem

Over 2,000,000 C/C++ Programmers:

Typically Use Coding First Approach

Taught Religious Fervor And Hacking By Universities

Extensive Libraries Available:

- Target Hardware
- DBMS Support
- Windows APIs

Increasing The Commercial use Of Ada: The Solution

Short Term	Long Term
• Provide Appropriate Libraries	• Fund University Undergraduate Programs In SW Engineering: Practical, Not Research Oriented, Curricula Regularly Rotate Appointments (No Tenure) Require Program Certification: Initial Continuing
	• Fund Experimentational Research: Same Design, Implement In Ada, C, C++ Same Problem, Develop With Multiple Methodologies Collect Comparative Metrics
• Pray, Pray, Pray	• Promote Publication Of Practical Articles, Books On Aspects Of SW Engineering: Not Theoretical Or Abstract Not Hacker/Programming Approach Emphasize Procedural Approach Define Standards, Principles, Practices, Procedures

Participant Briefing
Gary Kennedy
IBM Federal Systems Corporporation

DEFENSE INFORMATION SYSTEMS AGENCY
ADA SUMMIT
OCTOBER 19-20, 1993

IBM / FEDERAL SYSTEMS COMPANY
INDUSTRY PANEL
ADA POSITION PAPER

Gary F. Kennedy, Manager
FSC Software Engineering Process
Bethesda, MD (301) 493-1170

DISA / ADA SUMMIT CONTENT

- IBM / FSC OVERVIEW
- FSC ADA EXPERIENCE
- LESSONS LEARNED
- RECOMMENDATIONS

DISA / ADA SUMMIT
FSC OVERVIEW

- FSC POPULATION PROFILE:
  - 11,700 PEOPLE
  - 52% ENGINEERS AND PROGRAMMERS
  - 5 MAJOR SITES / LABS

- APPROXIMATELY 3 BILLION $ REVENUE

- SYSTEMS INTEGRATION AND DEVELOPMENT
  - -- LIMITED COMMERCIAL PRODUCT DEVELOPMENT

- PRIMARY CUSTOMER: FEDERAL GOVERNMENT

IBM / FSC

DISA / ADA SUMMIT
FSC ADA EXPERIENCE

- OVER 30 ADA DEVELOPMENT PROJECTS IN 8 YEARS
  - INCLUDING 22 ACTIVE PROGRAMS
  - 5 MILLION LINES OF DELIVERED ADA CODE
  - CURRENT PROGRAM BASE WILL BE PRODUCING APPROX. 1 MILLION LINES OF ADA CODE / YEAR
  - PRIMARILY LARGE, COMPLEX APPLICATIONS
- IBM AND FSC HAVE INVESTED HEAVILY IN ADA
  - COMPILER TECHNOLOGY
  - INTEGRATED ADA DEVELOPMENT ENVIRONMENTS
  - ADA TRAINING AND EDUCATION
  - PARTICIPATION IN INDUSTRY STANDARDS GROUPS
- ADA IS STRATEGIC TO FSC
  - CORE COMPETENCIES AND ASSETS
  - EVOLVED AND DEVELOPED BY EXPERIENCE
  - ENDURING AND OVERCOMING THE START-UP PAIN
    - INITIAL EXPECTATIONS OVERSTATED
    - IMMATURE TOOLS AND ENVIRONMENTS
    - SHORTFALL IN REAL-TIME FEATURES/BINDINGS

IBM / FSC

DISA / ADA SUMMIT
FSC ADA LESSONS LEARNED

- THE HARD PART (THE FIRST 5 YEARS)
  - SIGNIFICANT START-UP IMPACT
  - INITIAL PRODUCTIVITY DOWN
  - MATURE PRODUCTIVITY (NEW CODE) EQUALS OTHER 3GL'S DURING DEVELOPMENT PHASE
  - REAL-TIME APPLICATION IS MORE COMPLEX
  - SLOW EVOLUTION OF SUPPORT TOOLS AND INTEGRATED DEVELOPMENT ENVIRONMENTS
  - SIGNIFICANT INVESTMENT IN REAL-TIME FEATURES AND PRODUCT / INTERFACE BINDINGS
  - HARD TO QUANTIFY BENEFITS ON SHORT DEVELOPMENT LIFE-CYCLE ALONE

- THE GOOD RESULTS....
  - HIGHLY MAINTAINABLE CODE PRODUCED
  - IMPROVEMENT IN QUALITY ON INITIAL DELIVERY
  - REUSE, PORTABILITY IMPROVED
  - BASIS FOR COMPANY-WIDE ASSET REUSE
  - REENFORCES GOOD SOFTWARE ENGINEERING PRINCIPLES AND PRACTICES

IBM / FSC

DISA / ADA SUMMIT RECOMMENDATIONS

- FIRST AND FOREMOST...
  - FACILITATE INSERTION WITHIN ACADEMIA

E.G...

- GRANTS / STUDIES
- FREE COMPILERS
- FOCUS ON ADA LANGUAGE AS A FACILITATOR OF GOOD SOFTWARE ENGINEERING PRACTICE IN IMPLEMENTATION
- INCENTIVIZE ACADEMIA / INDUSTRY PARTNERSHIP

INDUSTRY NEEDS DISCIPLINED SOFTWARE ENGINEERS FOR TOMORROW, NOT JUST PROGRAMMERS

IBM / FSC

DISA / ADA SUMMIT RECOMMENDATIONS

- SECOND AND ESSENTIAL...
  - DOD AND THE GOVERNMENT NEEDS TO SEND A CONSISTENT MESSAGE

    I.E...

    -- MANDATE NEEDS TO BE APPLIED MORE CONSISTENTLY OR ELIMINATED

  - STREAMLINE/ACCELERATE EVOLUTION OF THE LANGUAGE STANDARD AND DEPLOYMENT

ONE OF THE MAJOR INHIBITORS TO COMMERCIAL ACCEPTANCE IS THE PERCEIVED DOD FLAVOR OF THE LANGUAGE AND THE EXPECTATION OF SLOW PROGRESS vs RAPID EVOLUTION ESSENTIAL TO FIELD COMMERCIAL PRODUCTS IN THE HIGHLY COMPETITIVE AND RAPIDLY EVOLVING TECHNOLOGY ENVIRONMENT OF TODAY

IBM / FSC

DISA / ADA SUMMIT RECOMMENDATIONS

- THIRD AND VITALLY IMPORTANT....
  - EXTEND MARKET OPPORTUNITY WITHIN THE GOVERNMENT

E.G...

-- MIS

-- NON-TRADITIONAL SYSTEMS APPLICATION

-- MAINTENANCE AND UPGRADE OF DELIVERABLES

I.E...

INCENTIVIZE INDUSTRY TO MAKE THE INVESTMENT IN LONG TERM ASSETS

-- THE BENEFITS OF ADA DEVELOPMENT ACCRUE TO THE 'OWNER' OF THE ASSET

-- THE MAJORITY OF PROGRAMS TODAY ARE DELIVERED TO THE GOVERNMENT TO BE MAINTAINED AND UPGRADED

IBM / FSC

DISA / ADA SUMMIT RECOMMENDATIONS

- FORTH AND FINAL...
  - ACCELERATE THE DEPLOYMENT OF ADA 9X
    - -- OO FEATURES WILL MAKE ATTRACTIVE TO COMMERCIAL APPLICATIONS
    - -- OO & RT FEATURES WILL RE-INCENTIVIZE FEDERAL APPLICATIONS

IBM / FSC

Participant Briefing
Robert Leif
Ada-Med

October, 1993 Dual Use

Use of Ada for Medical Devices

Robert C. Leif and Suzanne B. Leif
Ada_Med 1030 Mariposa Ave., Coral Gables, FL 33146.

Tel.(305)661-9048, Email CompuServe 76137, 2164

Health Industry Manufacturers Association, HIMA, Data

U.S. PRODUCTION

Item	1992	1995 Projected
Medical devices	$33.9 billion	
Diagnostic products	$5.5 billion	
Total Production	$39.4 billion	$51.0 billion
U.S. Industry Employment	267,000	

Health Industry Manufacturers Association, HIMA, Data

U.S. TRADE

Item	1992	1992 Change	1995 Projected
U.S. exports	9.0 Billion	up 13.1%	13.3 Billion
U.S. imports	4.9 Billion	up 11.5%	6.7 Billion
U.S. trade surplus (deficit)	4.1 Billion	up 14.6%	6.6 Billion
with the EC	1.56 Billion	up 9%	
with Japan	121 Million	up slightly	
with Canada	904 Million	up 45%	
with Germany	(116) Million	up 40%	
with Mexico	(70) Million	up 40%	
with the U.K.	257 Million	up 44%	

Health Industry Manufacturers Association, HIMA, Data

U.S. Medical Device Growth

Item	1992 Estimated	Projected Annual Growth Through 1995
Market size	up 8.3%	up 8%
Production	up 9.0%	up 9%
Exports	up 13.1%	up 14%
Imports	up 11.5%	up 11%
Trade surplus	up 14.6%	up 18%
Employment	up 4.5%	

FDA Authority:

The FDA is responsible for assuring the safety and effectiveness of medical devices under the 1976 Medical Device Amendments to the Federal Food, Drug, and Cosmetic act.

Both the FDA and the ISO 9000 require medical devices including software to be produced according to good manufacturing practices.

What the FDA Expects

Software Life Cycle

- Requirements
- Specifications
- Formal methodologies
- Modular software with structural testing
- Comprehensive configuration control
- Documented test cases
- Fail Safes Changes go back to early Phases
- Power up checks

Code

- **Written to a standard.**
- **Inspected and reviewed to ensure that it:**

Reflects the design

Follows the company policies

Certification

All medical devices which require a premarket notification 510(k) must submit a certification at the time of submission which is a:

"Written affirmation stating that software development was followed, that good quality assurance procedures were adhered to, and that test results demonstrate that the system specifications and functional requirements were met."

FDA Medical Device Good Manufacturing Practices
Manual 5th Edition, 1991

Validation of Manufacturing Equipment

Equipment: installation qualification

Process equipment should be:

- Reviewed
- Calibrated
- Evaluated
- Tested

to verify capability of specified operating limits.

FDA Medical Device Good Manufacturing Practices
Manual 5th Edition, 1991

“The manufacturer who has custom software prepared and validated by a contractor must ensure the software program is running properly and producing correct results before using the program to produce medical devices for distribution.”

Validation & WARRANTY

- Only Ada compilers pass a United States government approved validation suite.

Alsys warrants that the Compiler Programs and/or Executive Programs perform substantially as described in the Documentation and the Reference Manual for the Ada Programming Language.

Microsoft (Visual BASIC & C++) states, "LIMITED WARRENTY. Microsoft warrants that (a) the SOFTWARE will perform substantially in accordance with the accompanying written materials for a period of ninety (90) days from the date of receipt."

Borland C++ Limited Warranty

Specifically, Borland makes no representation or warranty that the software is fit for any particular purpose and any implied warranty of merchantability is limited to the sixty-day duration of the Limited Warranty covering the physical diskette(s) and physical documentation only (and not the software) and is otherwise expressly and specifically disclaimed.

Has FDA effectively mandated Ada without knowing it?

If so, we should inform FDA that their mandate can be fulfilled.

Beneficial Features of Ada

- Described by ANSI/MIL-STD-1815A accepted by ISO
- U.S. Government official reference manual
- Built in modern software engineering features
- General purpose language
- Portability
- Readability

  Pascal super set

  Self-documenting
- Safety

  Strongly typed

Range checking

Exceptions (map to a Subset of Hazards)

- Package Structure

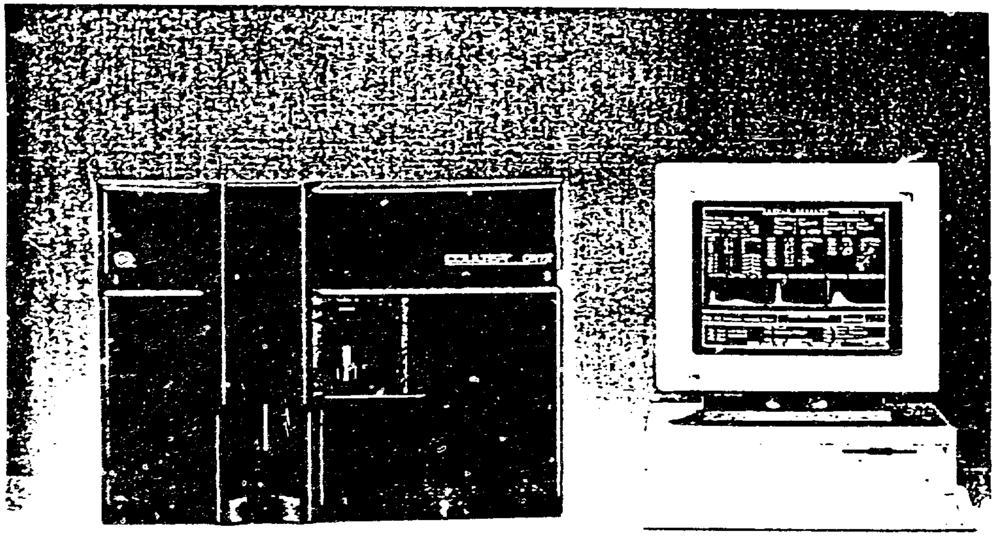
  Minimize visibility

  Track project schedule

- File management
- Tasking, real-time constructs, support for multiprocessors
- Established Methodology & Software Engineering Culture
- Independent Verification & Validation Organizations
- Software repositories
- Programmer Morale and Professionalism

The Coulter ONYX

- Midrange, quantitative, automated hematology analyzer
- For in vitro diagnostic use in clinical laboratories.
- Purpose is to separate the normal patient, with all normal system-generated parameters, from the patient who needs additional studies.



Unique Commercial Requirements

- Translatable User Interface
- Configuration Options

  User's Units

  Language
- Attractive User Interface
- Low cost platforms, PC 386 clone
- Microsoft DOS

Microsoft DOS

- This operating system is ubiquitous, inexpensive and provides a pseudo-embedded system.
- One major reason for the choice of Alsys Ada was because it produced 32-bit code for DOS.
- For true real-time performance, Ada tasking must directly control the disk.
- A subset of the POSIX specifications with DOS bodies should be developed. The same should apply to the Macintosh.

Software:

- Alsys 386 Ada compiler
- Supplemented with Aetech Ada_User libraries
- 120,000 lines of Ada code
- 3,470 Person Days
- 34.6 lines of code per day
- The Alsys 386 DOS compiler took 8 hours on a 25 MHz 486 to compile this code with optimization.

CONCLUSION:

- For a medical device, the benefits of Ada outweigh the disadvantages.
- The problem with Ada is that because of captive DoD market, it is technology rather than market driven.

Promoting the Use of Ada for Medical Devices

- Selling retread electrical engineers who like C is probably hopeless.
- Regulatory Compliance and Quality People will be more receptive.
- The U.S. Food and Drug Administration and the European Community Regulatory Bodies have the **POWER** to make it happen.

For more information,

call Ada_Med at (305) 661-9048

Email CompuServe #76137,2164

Participant Briefing
Ralph Crafts
Ada Software Alliance

Ada Dual-Use Workshop: Comments for the Industry Panel

Sheraton Premiere at Tysons Corner
October 19-20, 1993

By: Ralph E. Crafts, President
Ada Software Alliance, Inc. (ASA)
Rt. 2 Box 713
Harpers Ferry, WV 25425
304-725-6542
FAX: 304-725-6543
E-mail: SSTI@mcimail.com

Action Items for DoD/DISA to Increase Commercial Use of Ada

Three Major Categories:

1. Get serious about DoD's Ada directive

2.. Work more effectively with outside organizations

3. Market, promote, and advertise

Get Serious About DoD's Ada Directive

- Top execs must be clearly and forcefully committed
- Provide incentives (both positive and negative) that encourage the use of Ada
- Promulgate the "whys" of using Ada
- Put the onus of "proof" on the other languages
- Back up the words with funding and programs
- Revamp/eliminate debilitating standards

Work Effectively With Other Organizations

- Explore cooperative actions/programs with other government agencies
- Provide mechanisms for exploring & exploiting common needs
- Learn about the needs of commercial entities, and communicate at their level
- Facilitate cooperative efforts with Ada organizations and events (i.e., TRI-Ada, WAdaS, SIGAda, ASA)

Market, Promote, and Advertise

- Commercial successes are not often technically superior
- PR, marketing, promotion, and advertising are critical to commercial image and success
- The sales and marketing models are different
- DoD is not skilled in these areas--it will need help
- The Ada community also needs to be educated as to how to work with commercial markets

Summary

- Ada is still the best choice for accomplishing the DoD's mission--make that fact known
- The DoD must be serious about using Ada, and clearly show its commitment, to influence commercial interest
- Enormous benefits can be obtained by working with other organizations
- Marketing, promotion, and advertising are absolutely essential for success

Participant Briefing
Rush Kester
DC SIGAda

Ada Dual-Use Workshop

October 19, 1993

Rush Kester, Chair D.C. SIGAda

Ada as a Preferred Language

- Provide affordable retraining of existing staff in
  - Software Engineering
  - Object Orientation
  - Ada
- Publicize the availability of a production quality Ada compiler and run-time for existing mainframe platforms

- Convince decision makers by actions and words that DoD's commitment to Ada is strong
- Provide tools that ease conversion and re-engineering of legacy systems in other languages
- Make Ada the centerpiece of government research programs

- Promote the "better, faster, cheaper" attributes of Ada in non-Ada communities
- Convince Congress and non-DoD agencies that software policies should be based on the long-term economic advantage

Investments for Commercial Use

- Continue/increase funding of GNAT compiler and environment until sufficiently mature to make a good first impression
- Fund public domain translators from all popular programming languages
- Fund efforts to standardize bindings to all popular COTS products

- Support the commercialization of products developed for the DoD
- Support revisions in procurement regualtions that facilitate development of and with reusable software
- Fund Ada I.C. presence at non-Ada trade shows

Increase Ada's Appeal, Adoption, and Popularity

- Promote publication of Ada success stories and lessons learned
- Make Ada the centerpiece of government research and development programs
- Enforce the Ada mandate within the DoD and encourage other government agencies to adopt Ada

- Promote Ada technology transfer outside of the DoD
- Provide a supply of trained Ada software engineers and a demand for more
- Ada education in area high schools (Washington Ada Symposium (WAdaS))
  - Provide low cost training
  - Provide forum for technical exchange
  - Emphasize Ada successes and lessons learned in non-traditional areas

Participant Briefing
John Henry
EDSI

Ada Dual-Use Workshop

John Henry
EDSI

Expanding The Use of Ada in The Marketplace

- Market Ada to managers
- Make Ada desirable to programmers
- Get Ada into schools and colleges

Market Ada to Engineers

- Fund comparative studies of Ada and other languages
- Disseminate studies to widely read periodicals
- Kill the myths

Make Ada Desirable to Programmers

- Invest in tools and libraries
- Allow Ada to keep up with a changing world
  - C and C++ compilers change every nine months, Ada changed every 13 years (if we're lucky)

Make Ada Desirable to Programmers

- Subsidize a low-cost PC-based development environment
- Get serious about Ada reuse
  - Incentives to put code in libraries
  - Incentives to use code from libraries
  - Force DoD agencies to share software sources

Get Ada Into Schools and Colleges

- Subsidize a low-cost PC-based development environment
- Get a good author to write a useable Ada textbook
- Encourage Ada 9X as the standard and teach "object oriented programming with Ada", not "Ada, an object oriented language"

Participant Briefing
Richard Riehle
AdaWorks

Richard Riehle
Software Engineer
AdaWorks
261 Hamilton Avenue
Palo Alto, CA 94301
(415) 328-1815 FAX 328-1112
riehler@ajpo.sei.cmu.edu

Presented October 19-20 1993, Tyson's Corner VA

Introduction
Commercial Projects in Place

- □ NSTAR (Nippon Telephone & Telegraph
- □ Intelsat VII
- □ RADARSAT (Canada)
- □ Many International Projects
  - ○ China, Japan, Europe

Ada Dual-Use Workshop
Industry Track
Richard Riehle
rlehler@ajpo.sei.cmu.edu

Victim or Villain

"I'm not really bad. I'm just drawn that way."
Jessica -- *Who Killed Roger Rabbit*

"I'm not really bad. I've just been implemented that way."
Ada -- In a moment of wistful reflection

© AdaWorks™ 1993

Ada Dual-Use Workshop
Industry Track
Richard Riehle
rlehler@ajpo.sei.cmu.edu

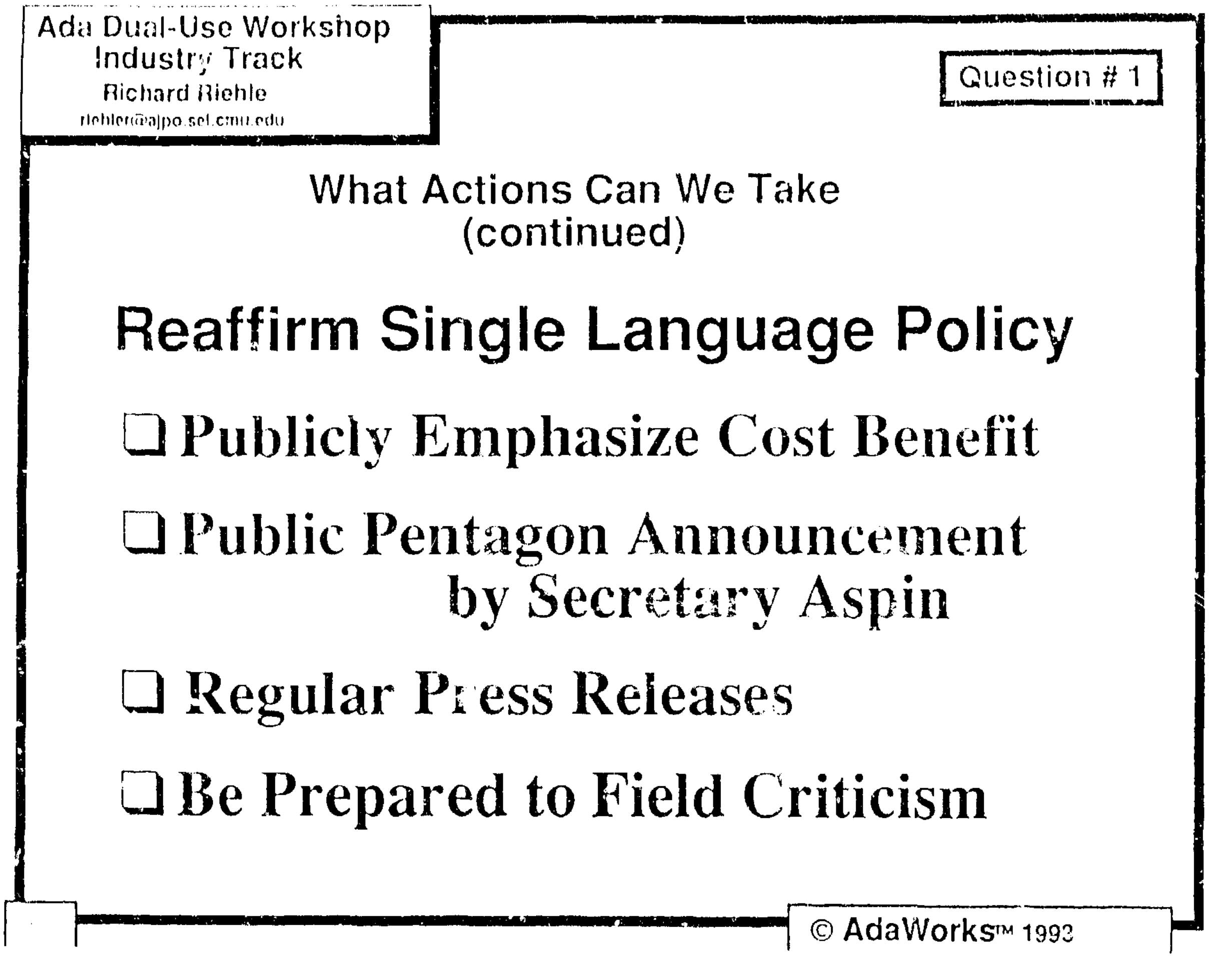
Question # 1

What Actions Can We Take

If "we" means the Department of Defense ...

- ❒ Reaffirm DoD Commitment
- ❒ Manage the Mandate
- ❒ Get all DoD Managers on-board
- ❒ Publicize at Highest Level
- ❒ Involve VP Gore's Cost Cutting Effort

© AdaWorks™ 1993



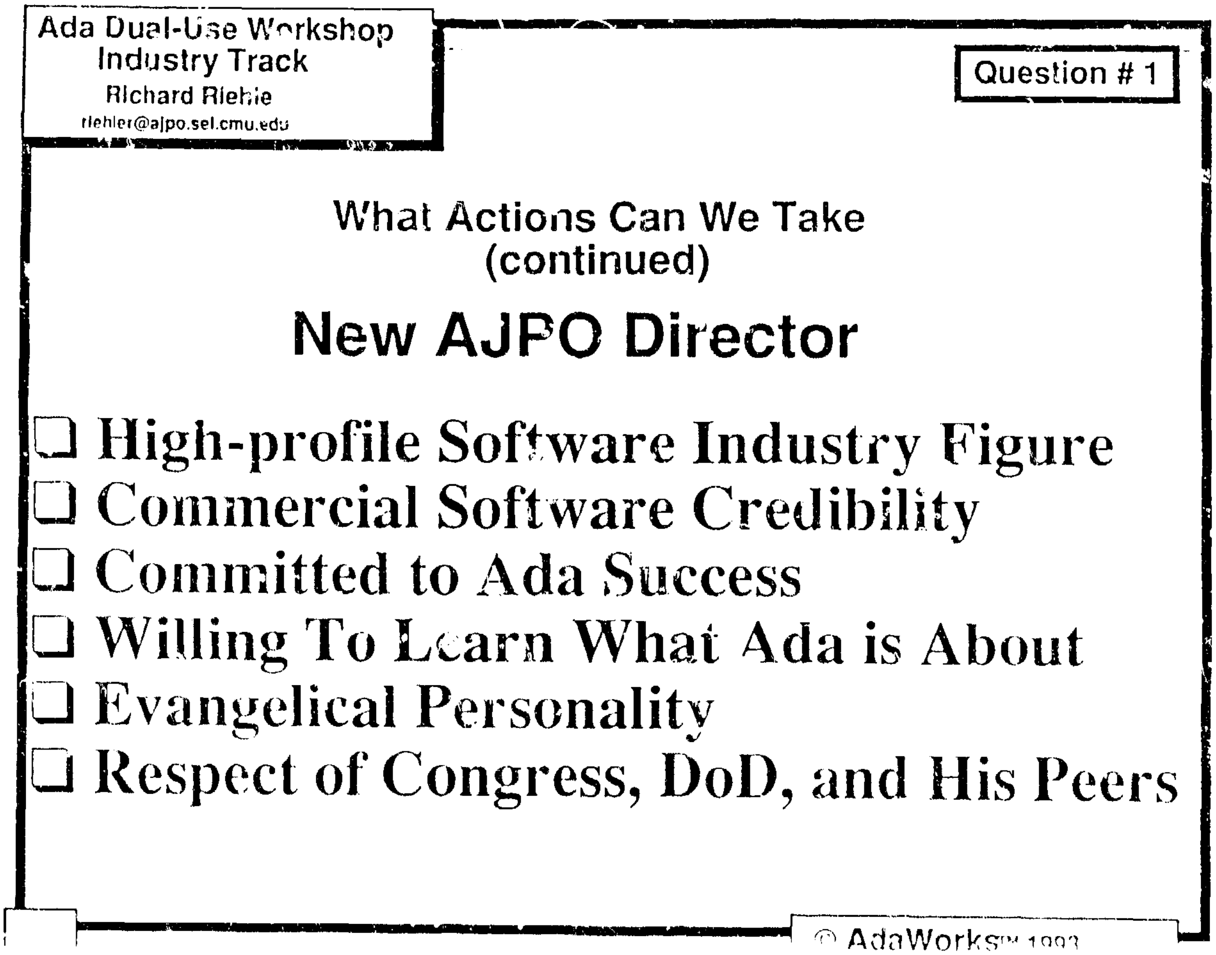
Ada Dual-Use Workshop
Industry Track
Richard Riehle
riehler@ajpo.sei.cmu.edu

Question # 1

What Actions Can We Take
(continued)

Reaffirm Single Language Policy

- ❑ Publicly Emphasize Cost Benefit
- ❑ Public Pentagon Announcement by Secretary Aspin
- ❑ Regular Press Releases
- ❑ Be Prepared to Field Criticism



Ada Dual-Use Workshop
Industry Track
Richard Riehle
riehler@ajpo.sei.cmu.edu

Question # 1

What Actions Can We Take
(continued)

New AJPO Director

- ❑ High-profile Software Industry Figure
- ❑ Commercial Software Credibility
- ❑ Committed to Ada Success
- ❑ Willing To Learn What Ada is About
- ❑ Evangelical Personality
- ❑ Respect of Congress, DoD, and His Peers

Ada Dual-Use Workshop
Industry Track
Richard Riehle
riehler@ajpo.sei.cmu.edu

Question # 1

What Actions Can We Take
(continued)

Place the Emphasis on Ada

- □ Enforce Waiver Process
- □ Establish Pilot Ada Projects
- □ More Emphasis on Training
- □ Sponsor Publication of Case Studies in Media
- □ Restore STANFINS-R

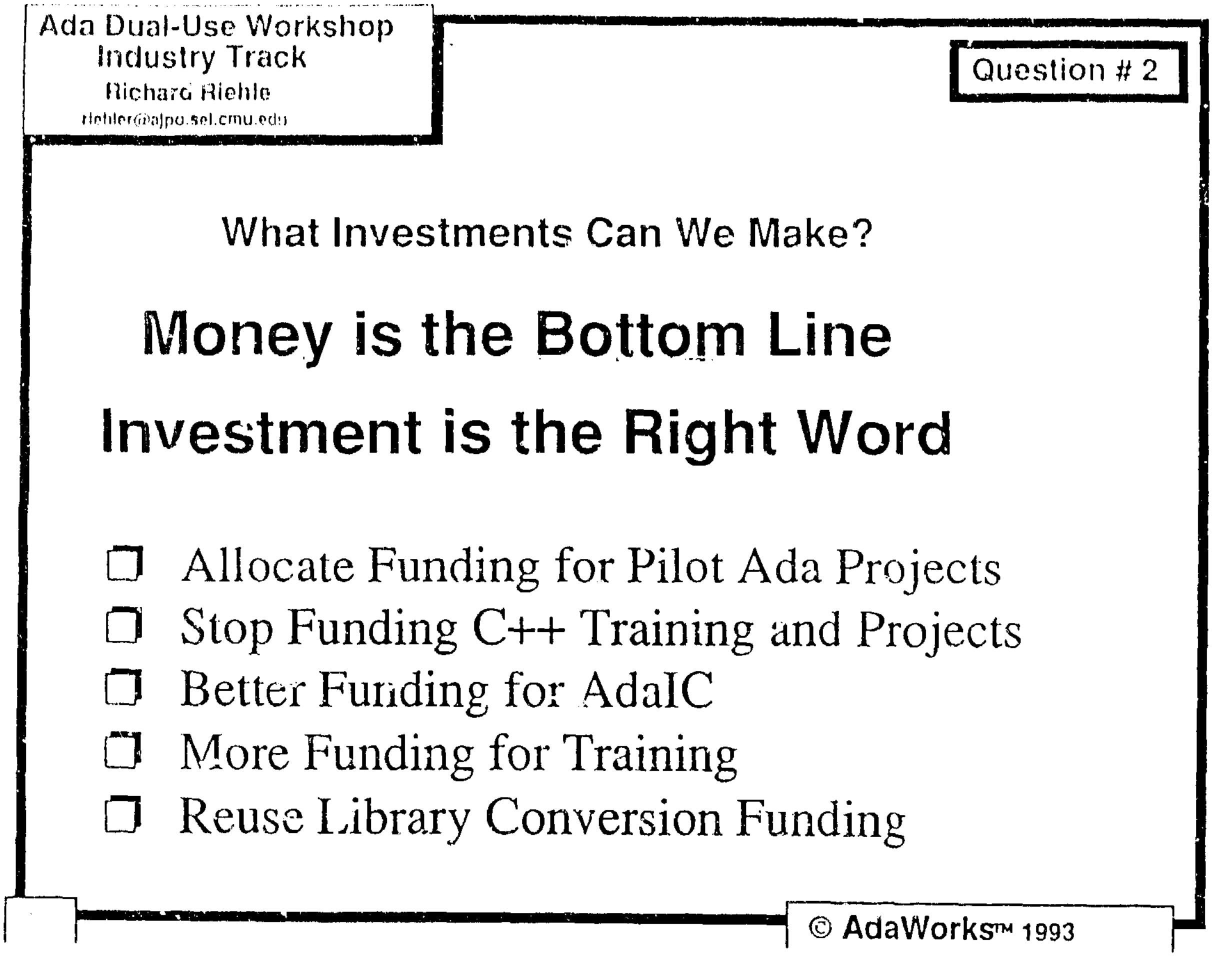
Ada Dual-Use Workshop
Industry Track
Richard Riehle
riehler@ajpo.sei.cmu.edu

Question # 1

What Actions Can We Take?
(continued)

Manage the Chain of Command

- □ USN blatantly ignores Ada
- □ USAF gets away with as much as possible
- □ Army is a little better
- □ USMC - Obeys Orders
  - ○ Salutes and Says "Yes Sir"
  - ○ Still knows who it works for
  - ○ Trains its personnel in Ada
  - ○ Uses Ada whenever possible



Ada Dual-Use Workshop
Industry Track
Richard Riehle
rlehler@ajpo.sel.cmu.edu

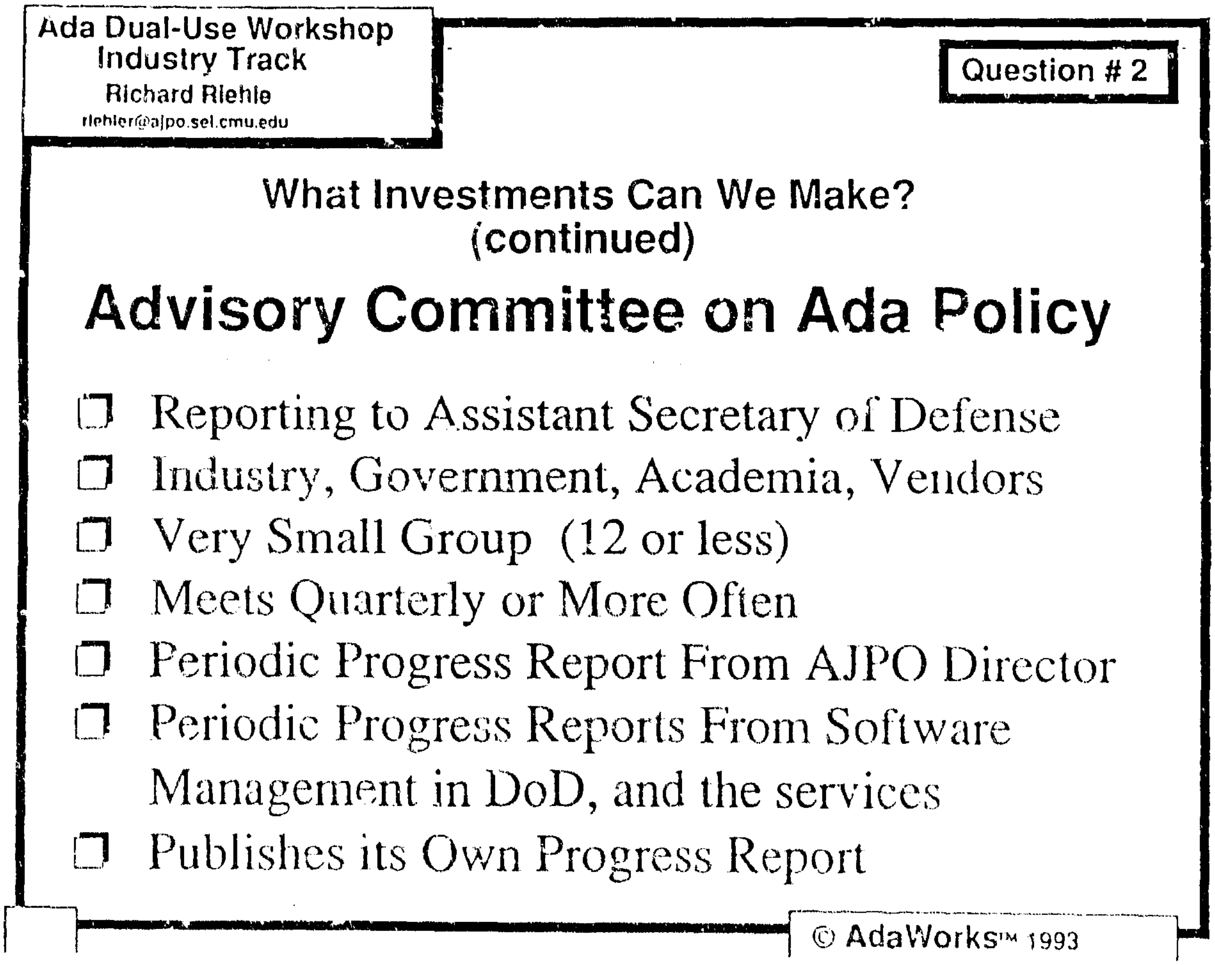
Question # 2

What Investments Can We Make?

Money is the Bottom Line

Investment is the Right Word

- ❐ Allocate Funding for Pilot Ada Projects
- ❐ Stop Funding C++ Training and Projects
- ❐ Better Funding for AdaIC
- ❐ More Funding for Training
- ❐ Reuse Library Conversion Funding



Ada Dual-Use Workshop
Industry Track
Richard Riehle
rlehler@ajpo.sel.cmu.edu

Question # 2

What Investments Can We Make?
(continued)

Advisory Committee on Ada Policy

- ❐ Reporting to Assistant Secretary of Defense
- ❐ Industry, Government, Academia, Vendors
- ❐ Very Small Group (12 or less)
- ❐ Meets Quarterly or More Often
- ❐ Periodic Progress Report From AJPO Director
- ❐ Periodic Progress Reports From Software Management in DoD, and the services
- ❐ Publishes its Own Progress Report

Ada Dual-Use Workshop
Industry Track
Richard Riehle
rlehle@ajpo.sei.cmu.edu

Question # 2

What Investments Can We Make?
(continued)

Commercial Market Research

- ❒ What Problem Are We Trying To Solve
- ❒ Who Makes Programming Language Decisions
- ❒ Language Selection Criterial
- ❒ Is Anti-military sentiment a factor
- ❒ Do we need more enabling technologies
- ❒ Do we need more tools
- ❒ What's the glamor of C++

No one currently involved in the Ada industry is qualified to perform a Market Research study in the commercial sector. This must be accomplished by an organization not currently involved in Ada projects, products, or marketing

Ada Dual-Use Workshop
Industry Track
Richard Riehle
rlehler@ajpo.sei.cmu.edu

Question # 3

What Can We Do To Increase Appeal?

Real Applications Development Tools

- ❒ Validation Is Not Enough!
- ❒ Plaform Specific I/O Packages
- ❒ Operating System Specific Packages
- ❒ Greater Portability
  - ○ MPE, AS 400, etc. -- no Ada
  - ○ Current Poor Support by IBM for VSAM and other OS/MVS/xx
- ❒ Ada with every hardware/OS shipment
- ❒ Simply Raise the Visible Commitment within DoD

Ada Dual-Use Workshop
Industry Track
Richard Riehle
rlehler@ajpo.sei.cmu.edu

Question # 3

What Can We Do to Increase Appeal? (continued)

Answer the Questions

- ❐ What makes a programming language popular?
- ❐ What is the role of compiler & tool Cost?
- ❐ Who chooses a language?
- ❐ Is Ada, alone, enough?
- ❐ What is the value of enabling technologies?
  - ○ e.g. Turbo Pascal, Quick C, Visual Basic, AdaSage
  - ○ Aetech, Alsys, Janus, and Meridian MS-DOS APSE
  - ○ SGI Graphics Environment Library
  - ○ XLib, Motif, and other windowing technologies

© AdaWorks™ 1993

Ada Dual-Use Workshop
Industry Track
Richard Rienle
rlehler@ajpo.sel.cmu.edu

Summary

- ❑ Department of Defense has enough software requirement to support a gigantic Ada industry.
- ❑ If the DoD truly supports Ada, the commercial marketplace will follow.
- ❑ Commercial software developers consider DoD Ada commitment to be something of a joke.
- ❑ Worse yet, many DoD software managers think it is a joke.
- ❑ What happened to the days when a commander gave an order, we snapped an acknowledging salute, and carried out that order?

© AdaWorks™ 1993

Participant Briefing
Fred Maymir-Ducharme
Unisys

Ada DUAL-USE WORKSHOP

Fred A. Maymir-Ducharme, PhD

Software Technology
UNISYS Government Systems
Reston, VA (703) 620-7559

fredmd@ajpo.sei.cmu.edu

19-20 October 1993

ACTIONS TO GET Ada ACCEPTED W/IN ORGANIZATION

- *COTS BINDINGS*
- *CASE/SEE SUPPORT*
- *INCREASE DEMAND FOR USE OF Ada*
- *SUPPORT/REWARD REUSE OF Ada COMPONENTS*
- *RECOGNIZE USE OF Ada DURING SCE/SPA*

Ada Dual-Use Workshop

INVESTMENTS TO INCREASE COMMERCIAL USE OF Ada

- *TECHNOLOGY TRANSFER*
- *EDUCATION & RESEARCH*
- *MARKET TO PUBLIC(NON-DOD) & PRIVATE SECTORS*
- *COTS / BINDINGS*
- *Ada9X AND Ada9X TRANSITION SUPPORT*
- *USE SCIENTIFIC / ADVISORY BOARDS FOR FEEDBACK*

Ada Dual-Use Workshop

TO INCREASE Ada's APPEAL, ADOPTION & POPULARITY W/IN INDUSTRY

- *FAIR/CONSISTENT MANDATES*
  - *LANGUAGE-INDEPENDENT REQUIREMENTS (Ada ATTRIBUTES)*
  - *REWARD PROPOSAL/USE OF Ada*
- *COTS/GOTS & STANDARDS BINDINGS*
- *MULTI-LANGUAGE SUPPORT / REENGINEERING TECHNOLOGY*
- *COORDINATE W/ SYSTEM/SOFTWARE ENGINEERING EFFORTS*
- *NON-DOD Ada SUBSETS/SUPERSETS*
- *EXPLOIT POTENTIAL SYNERGY BETWEEN Ada & OTHER TECHNOLOGIES*

Ada / DUAL-USE RECOMMENDATIONS

- *CONTINUE/INTEGRATE/EVOLVE AJPO ACTIVITIES*
  - *Technology Transfer (Ada Information Clearinghouse)*
  - *Technology Insertion (e.g., ATIP)*
  - *Technology Evolution (e.g., Ada9X, Ada20XX)*
  - *Exploit Synergy w/ other Software Technology Efforts*

- *Ada/REUSE*
  - *Use Reuse Libraries as Clearinghouses for Ada Components & Interfaces*
  - *Domain Specific Ada Component Interfaces*
  - *Foster Industry Consensus (e.g., publicly available interfaces)*

- *Ada/PROCESS*
  - *Ada Supports Sound Software Engineering - not evaluated in CMM*
  - *Advertise Successful Ada Process Models (e.g., CP-2044, CCPDS-R)*

- ***Support Emerging Industry Standards***

Participant Briefing
Barbara Johnson
Raytheon

Ada : A Contractor's View

Ada as Preferred Language

- Currently preferred but losing ground
- New technology creates complex problems
  - Object Oriented Development
  - COTS Integration
  - Expert Technology
  - Reuse
- Solutions needed now
- Training is always a required
- Ada not required on all DOD programs
- Cost is prohibitive

Ada : A Contractor's View

Increase Ada Commercial Use

- Additional access to DOD results & Information
- Consolidation of government reuse facilities
  - Internet access
  - Ada related tools
  - Ada bindings
- Incentives for commercial vendors
  - Monetary awards
- Low cost education

Ada - A Contractor's View

Increase Ada Appeal

- Early insertion into education process
- More cost effective
- More accessible
- More responsive to change

Participant Briefing
Shakil Kidwai
Electronic Data Systems

Ada

Management & Technical Issues in the Commercial World

Shakil Kidwai

BACKGROUND

- EDS is one of largest software developers
- Over 9 billion dollars in revenue
- DoD/Government business is only 9% of total revenue
- Presence in over 40 countries

MANAGEMENT ISSUES

- Management awareness of Ada capabilities
- Perception
- Availability of trained staff
- Demand by commercial customers

TECHNICAL ISSUES

- With introduction of CASE technology, language is no longer a key issue

- System will be maintained at specification level

QUESTION #1

- Send clear message
- Show commitment
- Issue no more waivers
- Enforce use of Ada
  - migration systems
  - new development

QUESTION #2

- Grants to universities
- Support/sponser Ada conferences
- Promote/advertise Ada success stories
- Provide incentive to vendors to use university resources

QUESTION #3

- Encourage vendors to develop COTS products in Ada
- inexpensive tools and compilers
- Promote reuse
- Share lessons learned

Participant Briefing
Karl Nyberg
Grebyn Corporation

Why I don't

Use

Ada

Karl A. Nyberg

Grebyn Corporation
P. O. Box 497
Vienna, VA 22183-0497

karl@grebyn.com
703-281-2194

Background

- Who
- What
- Why

Who

- Masters in EE & CS from MIT - 6/1981
- Employee #3 at Verdix - 1/1983
- Started up consulting business - 8/1985

What (Ada)

- Services
  - Ada Timesharing
  - Ada Verification Workshops
  - Proposal Evaluation - Marine Corps ATACC
  - Proposal Preparation - IDIQ AF Procurement
  - Proposal Preparation - Ada Verification
  - SEI Special Study on Ada's Input-Output
  - Technical Member, Ada 9X Requirement Team
  - Cost to Complete Evaluation
  - Compiler add-on support (X, POSIX, etc.)
  - SBIR Proposals
- Products
  - Ada Components
  - Ada Monthly (formerly INFO-Ada)
  - Approved Ada Language Commentaries
  - Ada : Sources & Resources
  - Annotated Ada Reference Manual
  - Ada Tapes
  - Ada CD ROM

Why Not Ada

- Services
  - Compiler economics
  - "Can't hire consultants"
  - "We can't possibly pay somebody that much who works out of their house"
- Products
  - Ada Components
  - Ada Monthly (formerly INFO-Ada)
  - Approved Ada Language Commentaries
  - Ada : Sources & Resources
  - Annotated Ada Reference Manual
  - Ada Tapes
  - Ada CD ROM

What (non-Ada)

- Defense Services
  - SDIO Denial of Service Analysis
  - Trusted MINIX development in C
- Non-Defense Services
  - UNIX Computer Security Consulting
  - Commercial process control software in C

Why (non-Defense)

- Services
  - Compensation is based on results
  - Get the job done

Impediments to Ada in DoD

- Captive market mentality
- DoD wanting to buy custom COTS
- Entry price to becoming DoD contractor
- Procurement policies supporting "buy on price, not on value"
- Adverse market influences

Impediments to Ada in Commercial Sector

- Ada success "products"
- Ada's military taint
- Up front investment
- Need to get to market
- Product Emphasis
- Competitive advantage

DoD v Commercial Sector[1]

DoD	Commercial Sector
Buy one	Sell many
One buyer	Many buyers
Contracted software	Product
Long term planning precludes quick reaction	Quick reaction to market forces
Limited market	Large market
Fixed profits	Large profit potential
Failures tolerated	Business Failures

[1] Taken from Gansler, "The Defense Industry," and Aharonian, "Microeconomic [illegible] for Ada Summit" as published in the latter and others

Strategies to promote commercial use of Ada

- Get DoD house in order
  - Tell; Ask; Pursue
  - Procurement policies based on long term value, not on short term price
  - Data rights issues
- Don't worry about popularity contest
  - DoD requirements are different
  - Ada designed to meet those requirements
  - Fear of Ada 9X failing to continue to do so
- Understand the free market economy
  - Support, encourage, seed
  - DON'T control, replace, meddle
  - The market does not always choose technical superiority

APPENDIX F

ADA DUAL-USE WORKSHOP
OCTOBER 19-20, 1993

VENDOR PANEL PROCEEDINGS

November 8, 1993

Defense Information Systems Agency
Joint Interoperability and Engineering Organization
Center for Information Management
Software Systems Engineering Directorate

TABLE OF CONTENTS

1.0 INTRODUCTION

This appendix contains the findings, conclusions, and recommendations for the three questions that were posed to the Vendor Panel. This data supports the summary of the vendor panel, as discussed in the Ada Dual-Use Workshop Summary.

2.0 QUESTION #1: **What can we [DoD and industry as partners] do to increase the market for Ada goods and services within the non-defense sectors?**

FINDINGS

The following findings were the result of participant briefings:

- Each of the Ada vendors is focusing on specific market segments, and striving to provide high quality cost effective solutions to customer needs in the chosen segments.

- Many DoD applications are dual-use, either in the specific sense that the exact product used by DoD is used by non-DoD customers (e.g., a medical instrument) or in the generic sense that DoD and commercial customers purchase similar products (e.g., aircraft flight control systems).

- DoD information systems are dual-use in the generic sense - DoD has unique requirements, but there are many similarities between DoD and commercial systems.

- Most information systems in both the DoD and commercial sectors are still written and maintained in COBOL.

- In pathfinding applications such as STANFINS, DoD has learned that Ada has significant advantages over COBOL for Information Systems applications.

- Ada 9X will provide significantly enhanced capabilities for Information Systems development, including object oriented programming.

CONCLUSIONS

Based on the session findings, the following conclusions were made by the panelists:

- DoD and Ada vendors can be effective partners in increasing the market for Ada goods and services in dual-use applications by focusing on specific market segments and capitalizing on their respective strengths.

- A particular target of opportunity is the reengineering of Information Systems into architecturally designed Ada components which can be ported easily to run in diverse environments ranging from mainframes to open client/server configurations and personal computers. Ada's technical benefits, especially portability and facilitation of reliable component level reuse, are especially valuable in these applications.

  The four highest priorities for DoD action are:

  1. Demonstrate a consistent commitment to use Ada for DoD applications.
  2. Partner with industry to promote awareness of Ada's benefits in targeted dual-use applications.
  3. Create a business environment which is conducive to private investment.
  4. Accelerate the availability of high quality Ada 9X products.

RECOMMENDATIONS

The following recommendations were derived from the findings and conclusions listed above:

- DoD should be aggressive in implementing the four actions listed above. A detailed analysis of each action area, including the findings which motivate the recommendation, conclusions, and specific implementation steps, is provided in the pages which follow.
- Vendors must also do their part. They must provide high quality products and services to meet DoD requirements, and they must be aggressive in promoting their products and services within non-Defense sectors.

Action Area 1: Demonstrate a Consistent Commitment to use Ada for DoD Applications.

FINDINGS

The following findings were the result of participant briefings:

- All vendor participants felt that Ada must be perceived as successful in the DoD sector to achieve a commercial market.

- The Ada vendor community unanimously rated DoD commitment to Ada and related success in the DoD software sector as the most important pre-conditions for success in the commercial sector.

- While DoD represents the Ada community's #1 reference account, DoD personnel are reluctant to appear to be promoting contractor products and services. This reluctance hinders the reference process that is an integral component of attracting commercial business.

- Vendor participants consistently cited the DoD's need to collect long-term project cost data as an important step in rallying the DoD behind Ada.

- DoD's inconsistent posture on Ada and its failure to follow through on its Ada mandate are serious impediments to Ada dual use. The DoD Ada mandate and level of commitment to Ada are not well communicated inside and outside of DoD.

CONCLUSIONS

Based on the session findings, the following conclusions were made by the panelists:

- DoD must make a clear consistent statement of its commitment to Ada and demonstrate that commitment through concrete actions.

- DISA should take the lead in the use of Ada 9X for Information System applications. Whereas Ada 83 was designed for embedded applications, Ada 9X has been extended with excellent support for information systems. Given DISA's mission and its central role in the Ada program, it must complete a pathfinding application of Ada 9X to Information Systems at the earliest possible date.

RECOMMENDATIONS

The following recommendations were derived from the findings and conclusions listed above:

- The DoD should prepare and distribute a succinct statement of why they are committed to language standardization and why they have selected Ada. This story should be told consistently across DoD.

- DISA should define a specific schedule to migrate its applications to Ada. At least one migration system should be converted this year. We suggest a budget of $2M - $3M this year, and recommend that the objectives of this initial migration effort (i.e., size and complexity of system) be established appropriate to that level of effort.

- The DoD needs to eliminate embarrassing incidents which impede commercial use of the language. A few of the examples cited are the following:
  - Ada not taught at the service academies
  - Military and civilian personnel ignoring policy, law, and directives with impunity
  - Lack of 6.1 and 6.2 funds for research supporting Ada
  - The selection of a COBOL finance system over Ada alternatives (STANFINS)

- The DoD should demonstrate its commitment through concrete actions
  - Public scrutiny for non-use
  - Penalties for non-use
  - Incentives for Ada use
  - Numerous announcements of specific systems which are being converted or implemented in Ada.

Action Area 2: Promote Ada Awareness in the Commercial Marketplace

FINDINGS

The following findings were the result of participant briefings:

- Ada is the best technical choice for creating applications that are reliable, maintainable, can evolve over a long period of time, can benefit from component level reuse, and must be rehosted to run on a variety of different hardware configurations during their life cycle.

- Ada is beginning to receive the attention and respect it deserves in the non-Ada community (for example, favorable references in technical papers and discussions in non-Ada conferences), but vendors would like to see a much higher level of favorable public awareness.

- Ada vendors would like to see wide dissemination of Ada success stories from DoD applications, especially applications which are "exciting". Vendors want to work with DoD to present such stories in a way which will be effective and understandable to the non-DoD community. The possibility of a negative reaction to DoD's involvement was discussed, but the consensus was that interesting and factual stories will be well received.

- Ada success stories are not widely disseminated for a variety of reasons, including the fact that 1) DoD personnel are reluctant to appear to be promoting contractor products and 2) commercial users are reluctant to disseminate information about their strategies and internal processes.

- About half the people attending the vendor session would be pleased to see DoD support market activities aimed at enhancing positive Ada awareness, such as advertising of Ada benefits and trade show participation. The other half of the attendees were convinced that DoD could be more effective at enhancing the commercial market for Ada through public relations activities (see above), and feared that advertising and trade show activities would be under-funded, and inconsistently executed. They feared, therefore, that the results would be disappointing.

CONCLUSIONS

Based on the session findings, the following conclusions were made by the panelists:

- DoD will benefit greatly if there is a widespread appreciation for Ada benefits and the business case for using Ada. These benefits will be in the areas of increased availability of products to meet DoD's needs, lower prices for the products and services DoD buys, and increased availability of Ada trained personnel to work on DoD programs. DoD and industry must work together to enhance Ada awareness.

RECOMMENDATIONS

The following recommendations were derived from the findings and conclusions listed above:

- Focus DoD's participation in Ada awareness activities on DoD's strengths - publicizing DoD's successful uses of Ada in a forceful and dramatic way. Specific approaches include:

  1. Have senior decision-makers highlight Ada's benefits and DoD's rationale for using it.

  2. Encourage DoD personnel and contractors to publish technical articles and give presentations at technical conferences which provide the facts and benefits of using Ada.

  3. Make it clear that it is appropriate for DoD personnel to be "references" for Ada products and services which have been used effectively on DoD projects.

4. Develop, in partnership with commercial companies, effective presentations of the business case for Ada.

5. Explain DoD's rationale for committing to Ada, to civilian government personnel, especially targeting personnel involved in joint programs with the DoD.

- Make it clear that DoD personnel are encouraged to work with vendors in enhancing the public awareness of Ada benefits, business cases, and success stories.

- Issue press releases on systems implemented in Ada.

Action Area 3: Create a Conducive Business Environment

FINDINGS

The following findings were the result of participant briefings:

- The Ada market is substantial! Ada tools market data suggests the current market for Ada compilers and tools is about $300M per year, and there are about 50,000 people now using Ada.

- While the Ada market is substantial, the overly optimistic estimates of an extremely large marketplace during Ada's formative years (initial estimates of total annual Ada software development and maintenance exceeding $100B were cited) may hinder efforts to attract venture capital funding to launch a dual-use initiative. See the Background Notes Section for further clarification on the structure of the Ada market.

- Vendors were generally enthusiastic about establishing a partnership with government to stimulate the dual use of Ada. There were, however, some diffeiences of opinion about government's proper role in this partnership (e.g., good customer, good reference, direct funder of specific products).

- Several vendors cited specific instances of practices by the DoD that they perceived as providing direct competitors to their product lines.

- Vendors wanted DoD to focus its investment dollars on standards and customer applications which meet specific DoD needs, not on the development of software products which compete with industry.

- The DoD was positively cited for its efforts to stimulate a vigorous Ada parts business through software reuse centers. Participants felt that government could foster this Ada parts business and that a robust parts business could aid in penetration of commercial sector applications.

- Many vendors suggested strategies of incentives for using Ada and penalties for not using Ada on DoD developments as a means for stimulating Ada usage.

CONCLUSIONS

Based on the session findings, the following conclusions were made by the panelists:

- The DoD should emphasize its role as an informed consumer of Ada goods and services.
- The DoD should be straightforward, open, and consistent about its procurement intentions for Ada products and for developing applications written in Ada. The vendor community will meet the Ada product and service needs of the DoD.
- The vendor community should lead the thrust of packaging Ada goods and services for use in the commercial sector.

RECOMMENDATIONS

The following recommendations were derived from the findings and conclusions listed above:

- DoD should focus first and foremost on its need to have its software written in Ada. The DoD should be rock-solid in its commitment to Ada and it should have a plan to implement all new systems in Ada and to transition legacy systems to Ada. The plan should be implemented and tracked, and progress should be periodically released.
- If someone will be writing DoD software, DoD should make sure that they have an Ada compiler and tools, including CASE tools that support development and production of quality Ada software. DISA's broad software process assessment capability should be used to monitor Ada usage within DoD Central Design Activities (CDAs) and at DoD developer sites.

Action Area 4: Accelerate the Availability of High Quality Ada 9X Products

FINDINGS

The following findings were the result of participant briefings:

- Ada 9X extends Ada with features which provide object-oriented capabilities and improved consistency of real-time performance.

- DoD personnel and contractors should use Ada 9X as soon as there are proven quality implementations. Ada 83 is an excellent tool, and there is no reason to rush the transition to Ada 9X. This attitude is reflected in the Ada 9X transition plan.

- Once high quality implementations of Ada 9X can be delivered, an excellent strategy for expanding commercial use of Ada is to position Ada 9X as a *new* language providing *safe reliable object-oriented* programming. This commercial positioning is somewhat inconsistent with the facts (Ada 9X is technically an evolutionary improvement on Ada 83, and upwardly consistent from the point of view of existing Ada applications), but the positioning will be useful in convincing commercial users to consider using Ada (9X).

- One way to reconcile the facts that Ada 9X is the best available tool for marketing Ada to the commercial community and that DoD should not use an Ada 9X compiler and tools until they are proven is to accelerate the availability of high quality Ada 9X tools.

- Ada compiler vendors estimate that the transition to Ada 9X will cost between $1M and $5M, depending on the number of hosts, targets, and tools they support, and various technical factors. There are about fifteen base compiler technologies which have been rehosted, retargeted, and integrated to create the hundreds of specific Ada products in the market today.

CONCLUSIONS

Based on the session findings, the following conclusions were made by the panelists:

- DoD should accelerate the availability of high quality Ada 9X implementations.

- DoD should provide monetary incentives combined with careful evaluation of Ada 9X products to accelerate the availability of Ada 9X products and to smooth the transition to Ada 9X.

- A DoD commitment of $10M would pay between 20% and 40% of the cost of migrating the base compiler technologies to Ada 9X
- Any DoD funding aimed at accelerating 9X availability s· ·uld be done as a partnership with vendors, and done in a way which incentivizes commercial investment in Ada 9X products.

RECOMMENDATIONS

The following recommendations were derived from the findings and conclusions listed above:

- Invest at least $10M this year to buy Ada 9X compilers, tools, and bindings for future delivery. This DoD investment will accelerate the transitions to Ada 9X and stimulate the recipients to make the additional investments to complete the products.
- Plan and implement early applications of Ada 9X within DoD. Goals are to validate the quality of the Ada 9X implementations, to create favorable references for Ada 9X technology, and to provide constructive user feedback to help the vendors improve their Ada 9X products. DoD programs will have to be funded to pay the cost of their leadership in the Ada 9X transition. The Ada Technology Insertion Program (ATIP) is a model for what is required, but the required funding will be three to five times greater than previous funding of the ATIP initiative ($10M-$15M).

3.0 QUESTION #2: **What investments should we (DoD and industry in partnership| make and initiatives should we take to increase the commercial use of Ada outside of defense applications.**

FINDINGS

The following findings were the result of participant briefings:

- It might seem possible to increase usage by giving product away. In the extreme, that would eliminate the market, since no revenue is generated when a product is given away.

- There was substantial discussion on the pros and cons of the government paying to have software developed and then giving it away. GNU Ada and AdaSAGE were particularly controversial. A few of the participants believe that it is never appropriate for the government to give software away. Most participants believed that it is appropriate for the government to give software away in specific situations, but that the danger of eliminating the incentive for private investment must be considered in the government's decision making process.

- Bindings (interfaces) to popular commercial database management systems, teleprocessing and network software, operating systems, and graphics user interfaces must be available as COTS products. Bindings are most efficiently implemented as features of the Ada system and the product being interfaced (i.e., the DBMS, TP and network software, OS, or GUI).

CONCLUSIONS

Based on the session findings, the following conclusions were made by the panelists:

- A majority of the participants were glad that GNU Ada is being created and will be available to increase academic interest in Ada. The participants did not have enough information to make any recommendation related to AdaSAGE. A vocal minority felt strongly that AdaSAGE products, thereby maximizing innovation and not providing free products that inhibit private investment.

- If DoD can adopt open domain specific architectures which are consistent with commercial practices and the requirements of non-DoD users, then Ada vendors can sell dual-use products, such as tools and reusable components, to both the DoD and commercial customers.

- DoD should incentivize vendors to provide Ada bindings to COTS products.

RECOMMENDATIONS

The following recommendations were derived from the findings and conclusions listed above:

- Implement all the recommendations provided in response to Question 1.

- Invest in the analysis and definition of standard open software architectures which both meet DoD requirements and meet the needs of as many commercial customers as possible. Develop these standards in partnership with industry. Incentivize vendors to provide quality tools and reusable components for these architectures.

- The highest priority recommended investments are as follows:
  - Continue funding for the Ada 9X Program as currently planned.
  - Invest in a DISA Ada migration plan and the near-term conversion of one migration system.
  - Joint investments by DoD and industry to develop Ada 9X compilers, tools, and bindings.
  - Increase funding for ATIP and focus specifically on Ada 9X.

4.0 QUESTION #3: What can we [DoD and industry as partners] do to increase Ada's appeal, adoption and popularity?

FINDINGS

The following findings were the result of participant briefings:

- All the findings, conclusions, and recommendations provided in answer to the first two questions are relevant to this question.

- Present Ada marketing efforts need to intensified.

- In the consumer market (i.e., desktop PCs), it would be too expensive to try to match the volume and sizzle of C and C++ promotions. The key is co-existence and careful targeting of Ada awareness activities to customers who can benefit most from Ada's technical advantages. On the product side, co-existence can be the path to making state-of-the-art "windows" environments available.

- Ada is often perceived as too tightly controlled by the DoD, even though it is an ANSI and ISO standard and there is significant non-DoD use.

- On the product side, a high priority is for bindings and interfaces which allow software written in C, C++, and other languages to be used as components of applications written in Ada.

CONCLUSIONS

Based on the session findings, the following conclusions were made by the panelists:

- We need to inject excitement and fun into Ada promotion. For example, Silicon Graphics takes Ada demonstrations to 80 conferences a year, and they are *fun.* (Only a few of these conferences are attended by the Ada community, so Silicon Graphics is delivering the message that Ada programs can be fun to a lot of new prospects.)

- Industry, in cooperation with DoD, must revitalize its efforts to increase Ada's share of the market for programming languages.

RECOMMENDATIONS

The following recommendations were derived from the findings and conclusions listed above:

- Implement all the recommendations provided in response to Question 1.
- In implementing the second recommendation, vendors (and perhaps DoD) should make a special effort to deliver the message that programming in Ada is state-of-the-art and fun.
- Where appropriate, it will be helpful if DoD can demonsrate high performance airplanes and other exciting systems which depend on software written in Ada for their performance.

5.0 PANEL

Panel members were selected to represent various facets of the industrial community.

	Name	Organization
Panel Chair:	Mr. William Carlson	Intermetrics, Incorporated
Panel Facilitators.	Mr. Paul Bacarro	Defense Information Systems Agency
	Dr. Bruce Burton	Intrermetrics, Incorporated
Panel Members:	Mr. Lee Ehrlichman	Tartan, Incorporated
	Mr. Phil Kiviat	Knowledgeware, Incorporated
	Mr. Mike Devlin	Rational
	Mr. Dave McAllister	Silicon Graphics
	Mr. Alan Olson	Digital Equipment Corporation

6.0 PARTICIPANTS

Participants were allocated a maximum of 15 minutes to give panel members their comments to questions within the vendor discipline. The following participants presented briefings during the Ada Dual-Use Workshop (Refer to Section 7.0 of this Appendix. Participant Briefings for copies of presentations). Participants are listed in the order of presentation.

Name	Organization	Briefing
Lee, Billy	Genesis Software, Inc.	Presentation - No Slides
Battey, Craig	Genesis Software, Inc.	Presentation - No Slides
Zimmerman, Bill	Intermetrics, Inc.	Presentation - Slides
Moskowitz, Paul	Verdix, Inc.	Presentation - No Slides
Thomes, James	Aetech, Inc.	Presentation - Slides
Taft, Tucker, S.	Intermetrics, Inc.	Presentation - Slides
Comer, Edward	Software Productivity Solutions	Presentation - Slides
Olie, Jean-Louis	ALSYS	Presentation - No Slides
Bond, Jan	Ventura Technology	Presentation - Slides/Position Paper
Thomas, William	Adirondack Computational Labs	Presentation - No Slides
Baker, Roger	Verdox Corporation	Presentation - No Slides
Houlihan, Paul	DDC-I, Incorporated	Presentation - No Slides
Need, Perry	Ventura Technology	Presentation - Slides
Banta, Richard	Westinghouse Electric Corp	Presentation - Slides
Dancy, Bonnie	EVB Software Engineering	Presentation - Slides
Dunbar, Terry	TLD Systems Limited	Presentation - No Slides
Maresca, Paul	Adasoft, Inc.	Presentation - No Slides
Lieherman, Wayne	Tartan, Inc.	Presentation - No Slides
Sapenter, Michael	Telos Corp.	Presentation - Slides
Hemdal, Goran	RJO Enterprises,	Presentation - Slides
Martinez, Carlos	Aetech, Incorporated	Presentation - No Slides

Note: Some of the views expressed were solely the participant's own and not of his/her organization.

7.0 PARTICIPANT BRIEFINGS

This section contains a copy of participant briefing materials provided to the Vendor Panel during the Ada Dual-Use Workshop. In cases where presentation materials were submitted in handwritten form, typed copies were created by the Ada Dual-Use Workshop staff. These copies are noted accordingly.

Participant Briefing

Bill Zimmerman
Intermetrics, Incorporated

Ada DUAL-USE WORKSHOP

Vendor Briefing on

Ideas and Suggestions of What DoD Can do to Expand the Commercial Use of Ada

October 19-20, 1993

William Zimmerman

Issues Affecting Commercialization of Ada

❑ The product vendors will invest and build Ada tools and COTS that are robust and competitively priced if a real market need exists

❑ The commercial users will probably embrace Ada if the products are competitive in price and performance and meet their requirements

❑ And both the commercial vendors and users are looking to the DoD for a real commitment to support and use Ada in all domains (embedded, business data processing and new technologies)

❑ But the DoD is looking to the commercial segments to be self sustaining

Ideas and Suggestions of What DoD Can do to Expand the Commercial Use of Ada

❑ Reaffirm Commitment to Ada

❑ Get the Word Out

❑ Get the Software Out

❑ Provide Incentives

❑ Seed the Market

❑ Inject Ada Into Industrial Standards

The DoD Must Show a Reaffirmation of its Commitment to Ada

❑ At issue is that the Ada community, especially the commercial sectors, are receiving mixed signals

❑ The DoD should establish consistency in its Ada mandate, specifically with regard to waivers

- Establish clear rules on when Ada shall be used (i.e., size of program, new vs. redesign, application/domains, expected life of system)
- And then enforce these rules

❑ Get key DoD officials, both military and civilian, to truly rally behind and embrace Ada, and for the right reasons... *because Ada is the language of choice for its inherent reliability and reduced life cycle maintenance costs*

❑ Resurrect and deploy the redesign of the Army's Standard Financial System (STANFINS-R), not only to show commitment to Ada, but to showcase a viable business application that would send a very positive signal to the commercial/business community

DoD Needs to Better Inform the Commercial Community and Get the Word Out By:

❑ Soliciting and conveying Ada success stories

- Especially outside of DoD and in other countries
- Require DoD contractors who develop in Ada to submit, along with their deliverables, a lessons learned report

❑ Publicize Ada through articles in popular trade journals (e.g., Datamation), as well as interviews, talk shows and even PBS. Establish statistics that would impress potential commercial users, i.e.,

- X% of ATC system around the world are written in Ada
- Y% of all commercial airlines fly software written in Ada

❑ Developing and distributing a Rationale Document targeted to the commercial sector clearly explaining and illustrating the benefits of Ada, to show how Ada is part of the solution

To Stimulate More Use of Ada, DoD Should Get the Software Out By:

❑ Better informing and encouraging Ada software re-use

- Focus on the contents of reuse repositories - the reusable components and architectures
- To encourage use, DoD should solve the problem of indemnifying users against potential legal problems associated with re-using someone else's software, per se, and for use in derivative works
- To enhance the effectiveness of software re-use centers, commercial Ada product information should also be made available

DoD Can Encourage the Choice of Ada by Providing Incentives to:

□ Industrial companies to develop applications in Ada

- Provide education and training in the understanding, use and benefits of Ada
- Offer and promote availability of software (via repositories)
- Offer consulting services in Ada project development
- Measure reliability of COTS products - the defect avoidance benefits of Ada will be apparent in the metrics

□ Commercial software and hardware suppliers to invest in Ada

- Only buy COTS products if they offer an interface to Ada
- Only buy computer hardware for development if Ada is installed

DoD Should Make Investments That Seed the Government Market by:

□ Providing funding for more new Ada starts

□ Placing large volume orders for commercial Ada COTS products

- We see lots of IDIQ contracts as vehicles to buy products, but we don't see lots of products being bought

□ Soliciting the support of Civilian government agencies to encourage the use of Ada

- Treasury/IRS
- Department of Justice
- Department of Transportation
- Department of Energy
- Others

DoD Should Inject Ada Expertise Into the Industrial Standards Groups Early On

❑ Traditionally its been standards first and Ada later (e.g., SQL, Windows)

❑ The Government should make a concerted effort to be sure there is some Ada expertise in the various industrial standards groups (e.g., OMG/CORBA)

❑ This could ensure the compatibility of industrial standards with Ada, thereby greatly facilitating the use of Ada in many commercial applications

Participant Briefing
S. Tucker Taft
Intermetrics, Incorporated

(Presentation materials were submitted in handwritten form. A typed copy was created by the Ada Dual-Use Workshop staff.)

Ada Dual-Use Workshop

T. Taft
Intermetrics, Inc.

October 19-20, 1993

Question 1

- Enforce mandate consistently, BUT...
  - Don't rely on the mandate.
  - Make the business case for Ada, both inside and outside the DoD.
- Business Case:
  - Reliability - Manager and Programmer
  - Expressiveness - Programmer
  - Commonality in training and reuse - DoD/CEO/CIO

Question 2

- Help to create a critical mass of Ada users:
  - Incentives to use Ada in high schools and universities.
  - Incentives to use Ada in DoD and non-DoD research labs.
  - Incentives to build COTS in Ada and/or interfaces to Ada.
  - Lower the entry barrier for "casual" use of Ada (e.g. GNAT)

Question 3

- Everything mentioned previously
- Ada 9X

Current Events

6/30/93	First Complete Ada 9X Draft RM Released (V3.0)
8/20/93	ISO WG9 Revision Workshop Concluded Successfully
10/1/93	ISO Committee Draft (CD) Balloting Begins (RM9X V4.0 plus updated 9X Rationale)
12/31/93	ISO CD Balloting Completes

Overall Schedule

October 1988	Ada 9X Revision Process started
March 1990	Mapping Contract awarded
Feb 1991	First Draft Mapping of Requirements to Revisions
April 1992	ISO WG9 Finalizes Core Mapping Specification
November 1992	ISO WG9 Finalizes Annex Mapping
August 1993	ISO WG9 Finalizes Draft RM
Fall 1993	Initial ANSI Canvass and ISO Ballot
Summer 1994	Final ANSI/ISO Balloting

Overview of Ada 9X

Full Support for Object-Oriented Programming

⇨ Ada 9X adds type extension (Inheritance) and class-wide programming (run-time Polymorphism) to Ada 83's existing support for Abstraction and Modularity.

Enhanced Support for Large Programs and Extensible Subsystems

⇨ Ada 9X supports a hierarchical library and multi-partition programs.

Full Support for Multi-Threaded/Multi-Processor Programming

⇨ Ada 9X adds protected types (for efficient Data-Oriented Synchronization) to Ada 83's existing Rendezvous-Oriented Synchronization.

Overview of Ada 9X

(continued)

Full Support for Information-Systems Applications

⇨ Ada 9X supports exact decimal arithmetic and International character sets, and through standard packages and pragmas, character and string handling, picture-oriented output, and Cobol interfacing.

A Much More Complete Standard Library

⇨ Ada 9X defines standard packages for Random Number Generation, Elementary Functions, Complex Arithmetic, Low-Level Task Control, Dynamic Priorities, Interrupt Handling, Per-Task Attributes, etc.

Ada 9X Business Case

Ada 9X is an Inherently Reliable Language

➔ Extensive compile-time error detection

➔ Thorough run-time error detection

➔ Very Strong Type Checking (Integers, Enumerations, Arrays, and Pointers are all strongly typed, with no implicit conversions between user-defined types)

➔ Dynamic Binding is easily recognized (only occurs when operands are of a class-wide type).

➔ Interfaces are checked and Information Hiding is enforced every time you compile (unlike checking based on separate tool)

⇨ Ada's thorough compile-time and run-time error detection automatically provides better detection for coding errors than a typical user-written test suite can.

⇨ Ada 9X programs will be more reliable when fielded, and errors will be caught earlier during the development cycle, where they are less expensive to fix.

Ada 9X Business Case

(continued)

Ada 9X Systems are Flexible and Safely Extensible

- ➔ Abstractions can be extended both by extending individual types and by adding child packages.
- ➔ Programs can be constructed dynamically from independently linked, cooperatively executing partitions, while retaining strong type checking across partitions.
- ➔ Synchronization primitives can be constructed efficiently using protected types as a building block.
- ➔ Dynamic Storage Allocation, Initialization, Assignment, and Finalization are all under programmer control.
- ➔ Ada 9X programs can easily interoperate with subsystems written in other languages, through shared data objects and calls (both call-outs and call-ins).

Ada 9X Business Case

(continued)

Ada 9X will be an ISO standard, and Ada 9X systems will be highly portable

- → ISO Standardization is well underway (at least a year ahead of C++).
- → Implementation dependences have been minimized in Ada 9X.
- → Additional bounds have been placed on implementation-dependent behavior.

Perspectives on C++

C++ is based on C, and inherits much of C’s inherent unreliability

- Pointer arithmetic substitutes for array indexing (no array bounds checking, and no array parameters, only pointers)
- Numeric types are all implicitly interconvertible (no way to distinguish a count of apples from a count of oranges)
- There is no integer overflow detection (it generally just wraps around)
- There is no null pointer detection (it just reads or writes page zero)
- The “==” vs. “=” problem remains in “if” statements
- The “missing break” problem remains in “switch” statements
- The confusing declarator syntax remains

Perspectives on C++

(continued)

A quote from a C++ user:

> "Hi, in this relatively small C++ project I'm working on, some very random weird errors started occurring and it eventually turned out to be accessing a function through a NULL pointer. Sigh. Anyway, it was quite difficult to track down, and *Deeper sigh* I just added a whole lot of code to the project and similar symptoms started occurring.
>
> ...Also, are there any 'safe programming' techniques which are used to avoid errors like this? Short of switching to another language :-)"

[From comp.lang.c++, 12 Sep 1993 (with permission of the author, who says...
"I DO like C++... just those *#&*! memory/pointer errors...").]

Summary

Ada 9X and C++ are essentially equivalent in expressiveness.

□ Ada 9X provides more sophisticated type composition (discriminant-dependent array components, access discriminants), while C++ has a more elaborate built-in inheritance mechanism.

□ Ada 9X provides a more sophisticated namespace control mechanism integrated with three-level visibility control (visible part, private part, and body), while C++ provides three-level accessibility control as part of the class construct.

But... Ada 9X and C++ differ dramatically in their inherent reliability features.

□ Ada 9X detects many errors at compile-time, and still-more at run-time

□ C++ has weaker compile-time, and no run-time checking.

Participant Briefing
Edward Comer
Software Productivity Solutions

Striving for Dual-Use Ada

and avoiding Null-Use Ada

Edward R. Comer
President
Software Productivity Solutions
122 Fourth Avenue
Indialantic, FL 32903
erc@sps.com

Be Sure to Maintain a Core Government Ada Market

- Without the core government-sector market, commercial interest in Ada would be nearly impossible
  - The current Ada US market of ~50,000 engineers (< 5%) is barely large enough
- Address Ada 9X uncertainties and risks
  - Upward compatibilities
  - Risks of a backward slide in compiler maturity
  - Compiler availability and commercial tool support
  - Funding for risk reduction and migration activities
- Address key Ada interface issues to keep pace with emerging COTS capabilities

Ada Must Better Support Heterogeneous Language Applications

- Provide better compilation system support for heterogeneous language applications
  - particularly C and C++
- Provide interfaces for easy use (reuse) of emerging COTS object-oriented software
  - GUIs, object bases, OO system software
- Fix the Ada policy to intelligently allow heterogeneous language applications

SPS

New Object-Oriented Software Encourages You Not to Use Ada

- GUIs
  - C callbacks, X-events, DLLs
  - instantiated C++ graphic object instances
- Object-oriented databases
  - C++ subclasses compiled in with the object base for application semantics
- Next generation object-oriented system software
  - CORBA (Common Object Request Broker Architecture): Interface Description Language (IDL) tools generates C++ headers to be compiled in with your application

Ada must accommodate the use of these new OO technologies with better interfaces and support for heterogeneous language applications.

SPS

Take a Market Approach to Commercial Ada Applications

- Identify high potential segments
  - Good product discriminators (e.g., real time, safety)
  - Sufficient market size and growth (e.g., process control)
  - Good competitive position (i.e., no firmly entrenched languages)
- Invest in dual-use technologies to give Ada a competitive advantage in key segments
  - Reusable software
  - Domain-specific architectures
  - Standard interfaces
- Do not try to compete in all segments
  - Do not sell Ada is as a “do it all” language
  - Focus on selected target segments

SPS

Participant Briefing
Jean-Louis Olie
ALSYS

Jean-Louis Olie
ALSYS

ADA DUAL WORKSHOP

1. What can we do to increase the market for Ada goods and services within the non-defense sectors?

To create a demand in new sectors such as transportation systems, telecommunication systems, medical instrumentation systems or management information systems, we need to make sure that the technical and, more importantly, business cases are made known. We need more success stories about the benefit of using Ada from DoD and industry.

2. What investments should we make and initiatives should we take to increase the commercial use of Ada outside of defense applications?

For the last ten years large investments have been made to develop the Ada technology and to have commercial off the shelf products available. Those products are today technically mature. To increase the commercial use of Ada, new investments are needed in the field of marketing to get the world to know that a commercial offering of Ada products exists today. Since the amount of spending that Ada vendors are doing to promote their products is directly related to the level of revenue in selling products, delays in funding DoD projects impact badly this effort. Since the case for Ada has been established, DoD and Defense industries should buy more COTS Ada products to allow Ada vendors to spend more in promotion. The increase in sales volumes will lead to cheaper products due to competitive pressure.

3. What can we do to increase Ada's appeal, adoption and popularity?

Ada is not visible enough outside of the Ada community. We have to make sure that Ada is represented in all conferences, exhibitions and trade shows who purpose is Software Engineering and not only Ada. Ada 9X is a new opportunity to talk about Ada.

Not enough people are trained to use Ada. To increase the number of skilled people we need to provide Academia and Training Institutions low cost but high quality and easy to use products.

We need to transform the initial "missionary approach" into a business oriented marketing approach.

Participant Briefing
Jan Bond
Ventura Technology

Ada Dual-Use Workshop

Position Paper

by

Jan E. Bond

COTS Vendor

1. What can we do to increase the market for Ada goods and services in non-defense sectors?

2. What investments should we make and initiatives should we take to increase the commercial use of Ada outside of defense applications?

3. What can we do to increase Ada's appeal, adoption and popularity?

Answer:

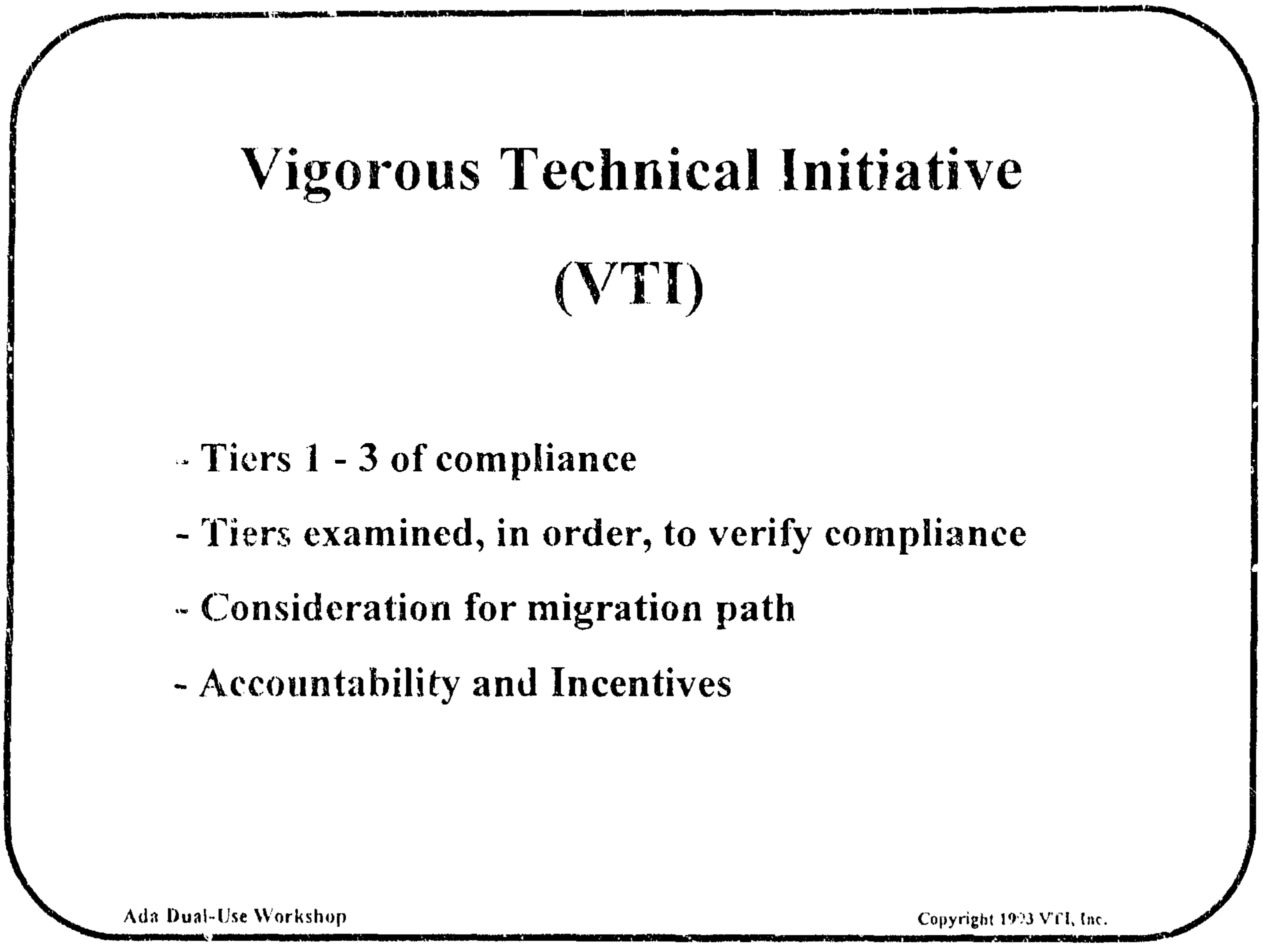
Demonstrated Commitment

Vigorous Technical Initiative

(VTI)

- X % written and maintained in Ada
- Y % written and maintained in 4GL
- where X > Y and X + Y >= 95
- Open System Environment (OSE) preferred
- Commercial RDBMS preferred

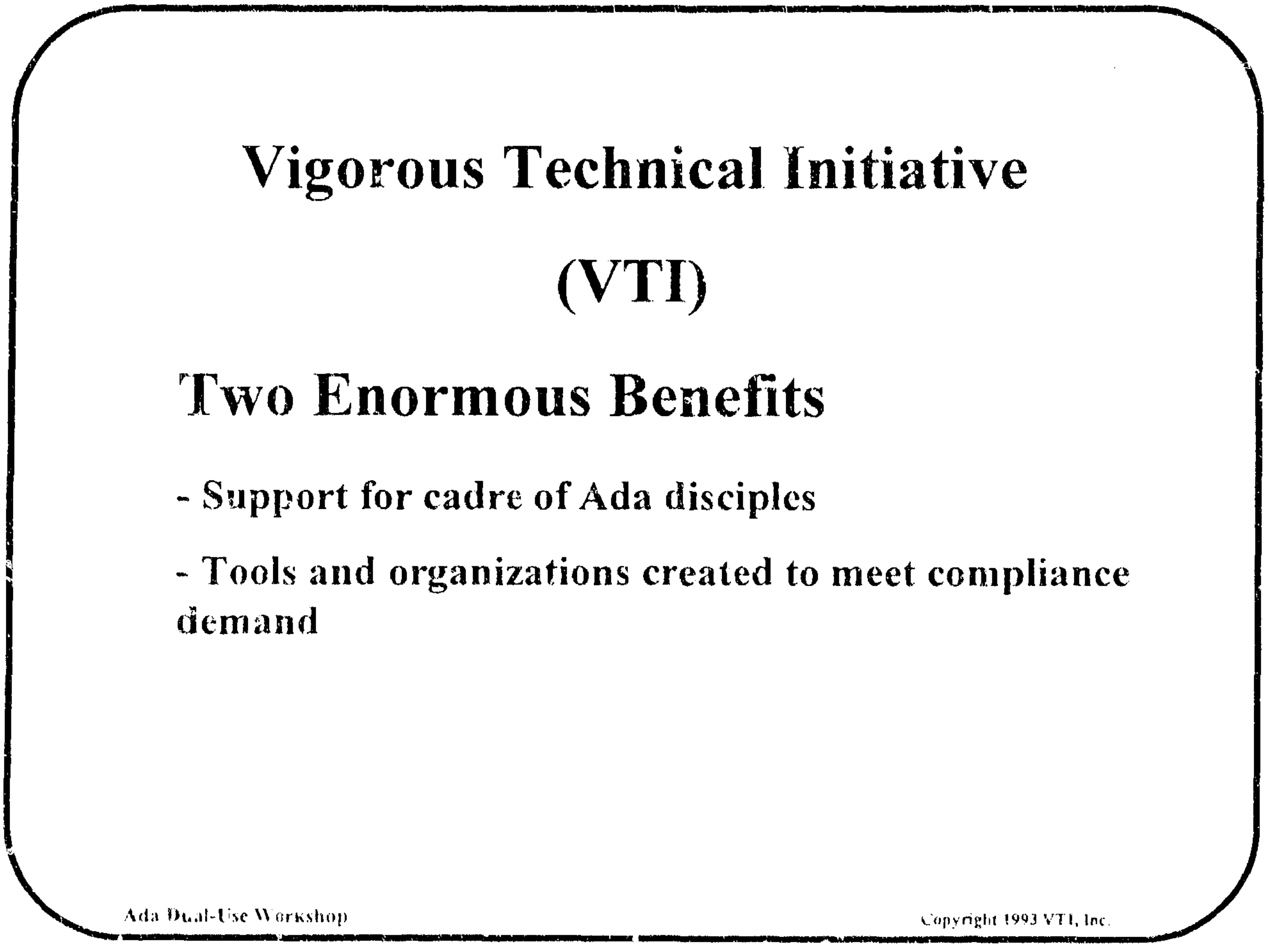
Applies to all COTS, Government Off The Shelf (GOTS), and Non-Development-Item (NDI) products.



Vigorous Technical Initiative

(VTI)

- Tiers 1 - 3 of compliance
- Tiers examined, in order, to verify compliance
- Consideration for migration path
- Accountability and Incentives



Vigorous Technical Initiative

(VTI)

Two Enormous Benefits

- Support for cadre of Ada disciples
- Tools and organizations created to meet compliance demand

Vigorous Technical Initiative

(VTI)

The pilot program for VTI in MIS has already been developed, tested, transitioned and killed.

Vigorous Technical Initiative

(VTI)

- The STANFINS/STARFIARS legacy
    - CASE repository
    - Program management tracking system
    - Suite of product Painters
    - Suite of code Generators

All of these were written in Ada!

Vigorous Technical Initiative

(VTI)

- The STANFINS/STARFIARS legacy
    - Specification 'painters' for screens, reports, etc.
    - Generation <u>from design</u> 75% SLOC
    - Documents <u>generated</u> from design repository

1. What can we do to increase the market for Ada goods and services in non-defense sectors?

2. **What investments should we make and initiatives should we take to increase the commercial use of Ada outside of defense applications?**

3. What can we do to increase Ada's appeal, adoption and popularity?

Answer:

Demonstrated Commitment

Vigorous Technical Initiative

(VTI)

The DoD must invest in the establishment of VTI.

1. What can we do to increase the market for Ada goods and services in non-defense sectors?

2. What investments should we make and initiatives should we take to increase the commercial use of Ada outside of defense applications?

3. **What can we do to increase Ada's appeal, adoption and popularity?**

Answer:

Demonstrated Commitment

Vigorous Technical Initiative

(VTI)

- Dispel Ada myths through education
  - CICS
  - CA/Datacom/Db
  - MVS
  - CPF (CICS Print Facility)
  - MVS
  - EBCDIC

Vigorous Technical Initiative

(VTI)

Demonstrated Commitment

- Field STANFINS and STARFIARS
- Migrate them to VTI compliance

Ada Dual-Use Workshop

Vendor Group
COTS Stakeholder Area

Position Paper

by

Jan E. Bond
VP of Product Development
Ventura Technology International

Ventura Technology International
7260 Shadeland Station, Suite 370
Indianapolis, Indiana 46256
317.594.1560
Fax 317.534.1562
Internet vti@netsys.com

I am VP of Product Development at Ventura Technology International (VTI). I have a strong Ada/MIS background based upon my experience as a key staff member on the STANFINS and STARFIARS Ada programs at Ft. Benjamin Harrison, Indianapolis, Indiana.

1. What can we do to increase the market for Ada goods and services in non-defense sectors?

Answer: Demonstrated Commitment

The DoD should establish that all software procured shall adhere to the Vigorous Technical Initiative (VTI). Procured software would include all Commercial Off The Shelf (COTS), Government Off The Shelf (GOTS) and Non-Development-Item (NDI) products. This approach reflects commitment to the DoD's own standards of software development. Specifically, all software should include:

- X % written and maintained in Ada
- Y % written and maintained in 4GL (where appropriate)
- where X > Y and X + Y > = 95
- where Open System Environments (OSE) are given preference
- and where commercial RDBMS use is given preference

The VTI should include tiers of compliance. The above describes Tier 1 compliance. Tiers 2 and 3 would have decreasing demands in each area. In order for an organization to procure software, Tier 1 must be examined first. If it can be demonstrated that no software exists at Tier 1 compatibility, Tier 2, and then Tier 3, products may be examined for procurement. There should be consideration given to a demonstrable migration path to compliance. Purchasing offices should be held accountable and incentives should be created for using products from Tier 1, then 2, etc.

The successful implementation of VTI will have two enormous benefits.

First, the DoD will be creating, expanding and most importantly supporting a large cadre of software disciples, competent in the very areas of software development and expertise that DoD has emphasized for its own software. This cadre exists today but has lacked support from the DoD.

An excellent example is Rational. They are a provider of a world class Ada/Software Engineering Environment and they have done everything in Ada.

Another example is Intermetrics, a company that uses Ada to develop software (its compilers) even in an Ada-hostile environment like IBM/MVS.

Neither of these firms made this decision because there was a mandate. The results are world class, maintainable software systems and more. Since these firms have first hand experience applying the language, they bring to their customers this expertise. This level of dedication to the DoD's own existing standard must be recognized and rewarded if Ada is to survive. Customers of Rational or Intermetrics get the benefit of a direct line to the knowledge and experience gained by implementing medium and large scale software programs in Ada. VTI will facilitate the rapid growth and cohesion that this small but talented group has lacked in the past.

Second are the tools, departments and even firms which will be created to meet the demands of complying with VTI. Some will develop home grown tools to assist in the transition. Others will seek outside assistance. In either case, these tools will make their way to the market and all will benefit. Firms will emerge from the transition with more maintainable, more marketable software systems and tool vendors will have an opportunity perhaps unparalleled to make an impact on their market.

The pilot program for VTI in MIS has already been developed, tested, transitioned and killed.

The STANFINS legacy is VTI. Computer Sciences Corp. was contracted to team with DFAS/Indianapolis to implement STANFINS, an Army financial management system, in Ada. Later this team was tasked to deliver STARFIARS, an Army inventory accounting system, in Ada as well. No MIS project with a scope similar to STANFINS had ever been attempted in Ada. In order to accomplish the task, the team developed four distinct supporting tools: a CASE repository, a program management tracking system, a suite of product painters (and analysis tools) and a suite of code generators. These tools were responsible for:

1) allowing analysts (non programmers) to 'paint' specifications for screens, reports, databases and etc

2) generating from design specifications 75% of the delivered source code that comprised STANFINS

3) generation from a central repository a complete set of DoD Std 7935A documentation

These tools gave the staff the near real-time access to the work-in-progress of their colleagues yielding the confidence to accomplish tasks such as changing architecture that affected entire systems/subsystems over a weekend. The energy created by enabling the staff to make such large changes in such a small amount of time led them to create more and better tools, to continue to improve the way the delivered systems worked, and to move with confidence to implement those improvements again and again.

And all of these tools were written in Ada.

As STANFINS headed toward the chopping block (not for technical reasons) some of the staff who saw the great value of and market potential for the tools used in developing STANFINS came together and formed a company exclusively to create a product which brings this Software Engineering experience to the market.

This story can be written again and again if the courage and vision exists to implement VTI. The DoD has the opportunity to create many STANFINS and an enormous number of startups in support of pursuing a standard which will take DoD and the market into the future.

2. What investments should we make and initiatives should we take to increase the commercial use of Ada outside of defense applications?

Answer: Demonstrated Commitment

The DoD must invest in establishing the VTI. Although not free, the cost will be reasonable. VTI stands ready to assist in this effort. We will participate in any way that we can. We feel this is an important endeavor for DoD, the government and the profession at large.

3. What can we do to increase Ada's appeal, adoption, and popularity?

Answer: Demonstrated Commitment

The DoD must stand by its commitment to Ada STANFINS and STARFIARS are the largest Ada/MIS systems ever attempted and need to be put forward by the DoD as examples of how Ada can be applied to non-embedded applications. These systems must be thrust into the limelight and fielded today with an eye towards compliance with VTI. The DoD must take two actions to make this happen. First a decision must be made to move forward. Second, resources need to be applied.

These systems should be transitioned to an OSE as soon as possible from their current proprietary platform. DFAS is already moving toward an OSE solution for STARFIARS. VTI stands ready to assist in any way to facilitate the transition and fielding of these systems.

DoD must move to reduce the cost of using Ada to develop software. The free Ada9x compiler is a first step but many, many more must be taken. Ada will continue to be a hard sell until the costs associated with using it approach those of COBOL.

There are no technological issues associated with moving from COBOL to Ada.

Time after time visitors came to the STANFINS site with preconceived notions about Ada and her (in)ability to be used for their programs. And in every case we were able to demonstrate Ada being used. One visitor came in saying, "We can't use Ada because Ada won't talk to CICS (an IBM/MVS base TP monitor)". Of course that argument fell apart when we executed Ada programs that had been talking to CICS for some time.

Of course these stumbling blocks turned out to be stepping stones that allowed the STANFINS team to grow. Here is a partial list of stumbling blocks turned stepping stones encountered on STANFINS:

CICS
CA/Datacom/Db
MVS
CPF (CICS Print Facility)
MVS Internal Reader
EBCDIC

In each of the above cases and many more, Ada gave the staff the power to prevail.

In conclusion, I submit that the technical case for Ada has been made time and again. The missing component is **commitment**. DoD must stand up and renew its commitment to Ada. Revitalize your commitment to the mandate and strengthen your overall commitment to Ada by adopting VTI today.

Participant Briefing
Perry Need
Ventura Technology

Ada Dual-Use Workshop

Vendor Group
CASE Tool Stakeholder Area

Position Paper

by

Perry Need
VP of Operations
Ventura Technology International

Ventura Technology International
7260 Shadeland Station, Suite 370
Indianapolis, Indiana 46256
317.594.1560
Fax 317.594.1562
Internet vti@netsys.com

1. **What can we do to increase the market for Ada goods and services in non-defense sectors?**

2. **What investments should we make and initiatives should we take to increase the commercial use of Ada outside of defense applications?**

3. **What can we do to increase Ada's appeal, adoption, and popularity?**

In answer to these questions several issues come to the forefront . These issues are:

What is the market
What needs to be done
What can I do, as a CASE vendor
What can DOD do

What is the market

Ada was explicitly made to address the software crisis. That is Ada's strength and that is Ada's market. Ada's market, at least the Information Systems (IS or MIS or DP) portion of it, consists of a rapidly growing number of Information Systems that are impossible to maintain. Furthermore not only is the number of Information Systems growing but the systems themselves are also growing in size and complexity.

It is worth noting that the software crisis, the problem that Ada was created to help solve, is not in the nature of a one-time fix like working down a backlog. The real nature of the problem is two-fold:

* how to produce bigger, better, maintainable, and cheaper systems;

* how to migrate the current inventory of aging and unmaintainable systems of today to the bigger, better, maintainable, and cheaper systems of tomorrow.

This is no small job. The market is huge. [**$2,300,000,000,000 in COBOL maintenance ** SLIDE] This is the market for Ada goods and services. More accurately this is just the Information Systems part of the market.

What needs to be done

There are two ways of bringing Ada to bear on this market. Both are equally important, both are critical to each other's success, and when both are combined they make Ada so powerful as to be unbeatable. These two ways are:

1) using Ada to build development support tools such as CASE tools (eg ASSERT) and APSEs (eg Rational), and

2) using Ada as the implementation language for Information Systems.

The use of Ada-built tools to build Information Systems in Ada has already been demonstrated. The STANFINS project of DFAS, an Ada system made by using Ada-built tools, achieved staggering productivity gains in the life cycle of the project. Subsequent use of the tools and lessons of STANFINS on the STARFIARS project replicated and improved the accomplishments.

STANFINS was done by a staff of COBOL experienced developers using Ada for the first time. Some of the results of that first time development and use of Ada-built tools to make an Ada-based Information System are:

* **6 times the industry norm for lines of code production (SLOC per staff-month), and**

* **from a common repository the:**

> * **generation of over 75% of 2 million lines of Ada code directly from design specs,**
>
> * **generation of a full suite of DoD 7935 documentation,**
>
> * **the support of all life cycle tasks.**

However the really big results of using Ada have not been shown yet. **The really big results of using Ada would show up in the maintenance portion of a project's life cycle.**

Unfortunately STANFINS was not deployed so its life cycle was not continued long enough to get a good measurement of maintenance benefits. But based upon the demonstrated results and the maintainability that was evidenced in STANFINS's software acceptance phase, there is every reason to believe that **the maintenance benefits would be even more staggering than the benefits experienced in the development phases of the life cycle.**

When results like these are achieved then it shows that Ada does fulfill its promise when used to do what it was made to do. **It also shows what happens when the Ada mandate is truly enforced and the Ada vision is vigorously embraced.**

The promise of the Ada vision has been proven to be real. The size of the market promises huge rewards for putting Ada to work doing what it was made to do. What needs to be done is clear:

ENFORCE THE MANDATE

EMBRACE THE VISION

What can I do as a CASE vendor

Use Ada. Embrace the vision.

If Ada remains unused then the vision will never be attained.

What I have done, as a software developer, is use Ada. I first used Ada because the DOD made me. When the DOD enforced the Ada mandate for STANFINS I had a clear choice: either embrace Ada and get the job done, or find another job. I embraced Ada to get the job done.

Now, as a businessman, I use Ada by choice. Knowing the results of using Ada, and looking at the market, I became a COTS vendor of Ada-built CASE tools. I want to get to the non-defense market because I have an incredible competitive advantage from using Ada. All I need is a track record of solving big IS problems and a supply of business cases and economic analyses.

What I will do, as a CASE vendor and a software developer, is **commit to using Ada in a big way**. We challenge the rest of the industry to make the same commitment. Some have, and some haven't.

The second issue to consider is that I, or any vendor, must bring to the market capabilities that are **comprehensive**, **flexible**, **affordable**, and **powerful** enough to do the job. To offer these capabilities I need two things: CASE tools and the expertise to use them.

Thanks to DOD, and STANFINS, I have Ada which has been proven to be a superior tool-building as well as an implementation language. Additional I need some other advanced technologies like 4GL and powerful RDBMS but again, thanks in large part to the DOD, these technologies are readily available. By using these technologies I produce capabilities that are comprehensive, flexible, affordable, and powerful.

The expertise to use Ada in solving IS problems is another matter. The expertise that is needed is in two tightly related areas: tool-users and Ada users. The same things can be said for both. When there are Ada jobs people learn Ada. When vendors use Ada technology they transfer the technology to their clients. When the number of pragmatic Ada users reach critical mass then significant problem solving will occur and the big job will get done.

What can DOD do

The single most important thing DOD can do to help the market is to focus on tackling the real job and avoid distractions. As the largest single aggregate Information System shop in the world DOD faced up to the problem long before the non-defense sector did and designed a tool to tackle the problem. DOD tested the tool and it worked. But DOD hasn't put Ada to work in a real way to solve the big problem.

DOD needs to **put Ada to work in a big way by enforcing the Ada mandate across the board.** The mandate has not been enforced and during that time the real problem of the software crisis has not gone away and in fact continues to grow. The failure of the non-Ada market over the last 15 years to produce any solutions powerful enough to handle the real problem indicates that none are forthcoming. Enforcing the mandate would at the very least force DOD Information Systems developers to come to grips with the various principles and practices of software engineering.

If the mandate is not enforced across the board then at least incentivize the use of Ada by giving funding priority to projects using Ada and Ada-based tools.

The second important thing DOD can do is to build an increasing community of Ada user expertise. We need significant projects that are held accountable for results, NOT protected, sheltered pilot programs.

When tool-users are held accountable for results then tools are seen as vital. Consequently the tool-builders are accountable to the tool-user. As shown by STANFINS when tool-builders are responsive to the needs of tool-users then tools are made that enable the users to get the job done. Useful tools must be flexible, adaptable, comprehensive, affordable and maintainable. When faced with an opportunity to make money by providing these kind of tools then vendors will either use a suitable language, i.e. Ada, or miss the market. If DOD were to nudge them along that path by requiring some significant amount of the tools be in Ada and that the tools support Ada then so much the better. In the course of making Ada-based tools and supporting Ada-based projects vendors will develop Ada-based expertise in solving big IS problems. Big IS problem-solving expertise is what is desirable to the non-defense market regardless of the language.

Get serious about the size and nature of the job by:

Using CASE tools to put the power of automated development support into the developer's hands to enable them to do their job.

Using COTS tools so that tools are built and supported by commercial toolbuilders thereby allowing project resources to be dedicated to making systems rather than split between project and tool support.

Require tools to use Ada thereby increasing the cadre of Ada expertise.

Make it easy to get to Ada-built and Ada-supportive tools by putting them on contract vehicles.

To move Ada from being from undesired to desired we must:

Deploy STANFINS so the maintenance benefits of Ada can be documented beyond any shadow of a doubt.

Document and publicize big Ada success stories by producing business cases and economic analyses much like the GAO does.
This should help to:

Jispel the Ada myths.

Remove Ada's stigma of being an intellectual and high-tech tool by putting Ada to work in the IS world.

Increase the number of Ada cadre experienced in solving big IS problems.

MARKET

"This planet has so far invested

$2,300,000,000,000

on the maintenance of **COBOL** code"

- Computer Weekly

Results of first time use

Ada tools on Ada IS

- 6 times the industry norm (SLOC/STAFF MONTH)
- from a common repository:
  - generation of over 75% of 2 million lines executable Ada
  - generation of full suite of DoD 7935 documentation
  - support of all life cycle tasks
- staff Ada illiterate at start

Ada fulfills its promise

because

the MANDATE WAS ENFORCED

and

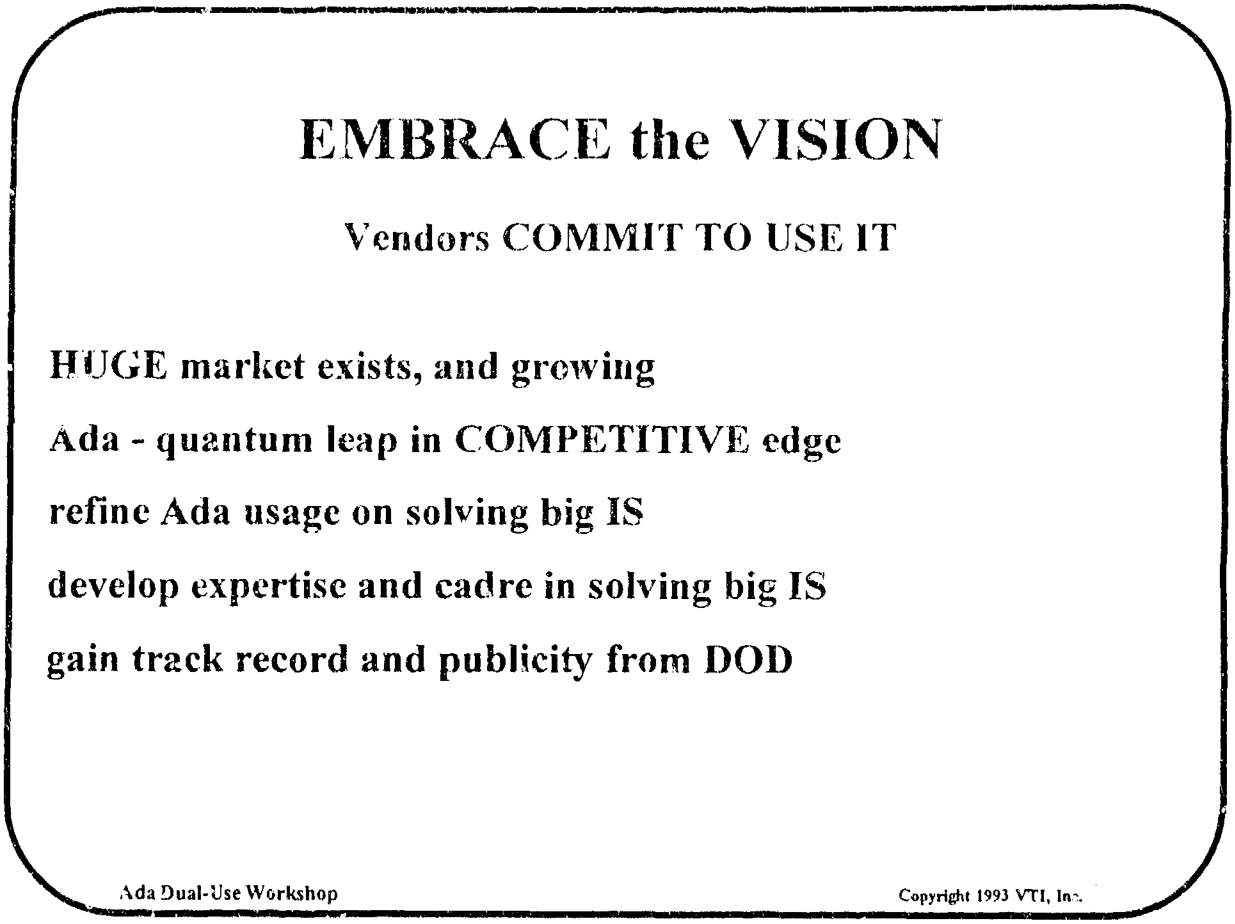
the VISION WAS EMBRACED

DOD

ENFORCE the MANDATE

Vendors

EMBRACE the VISION



EMBRACE the VISION

Vendors COMMIT TO USE IT

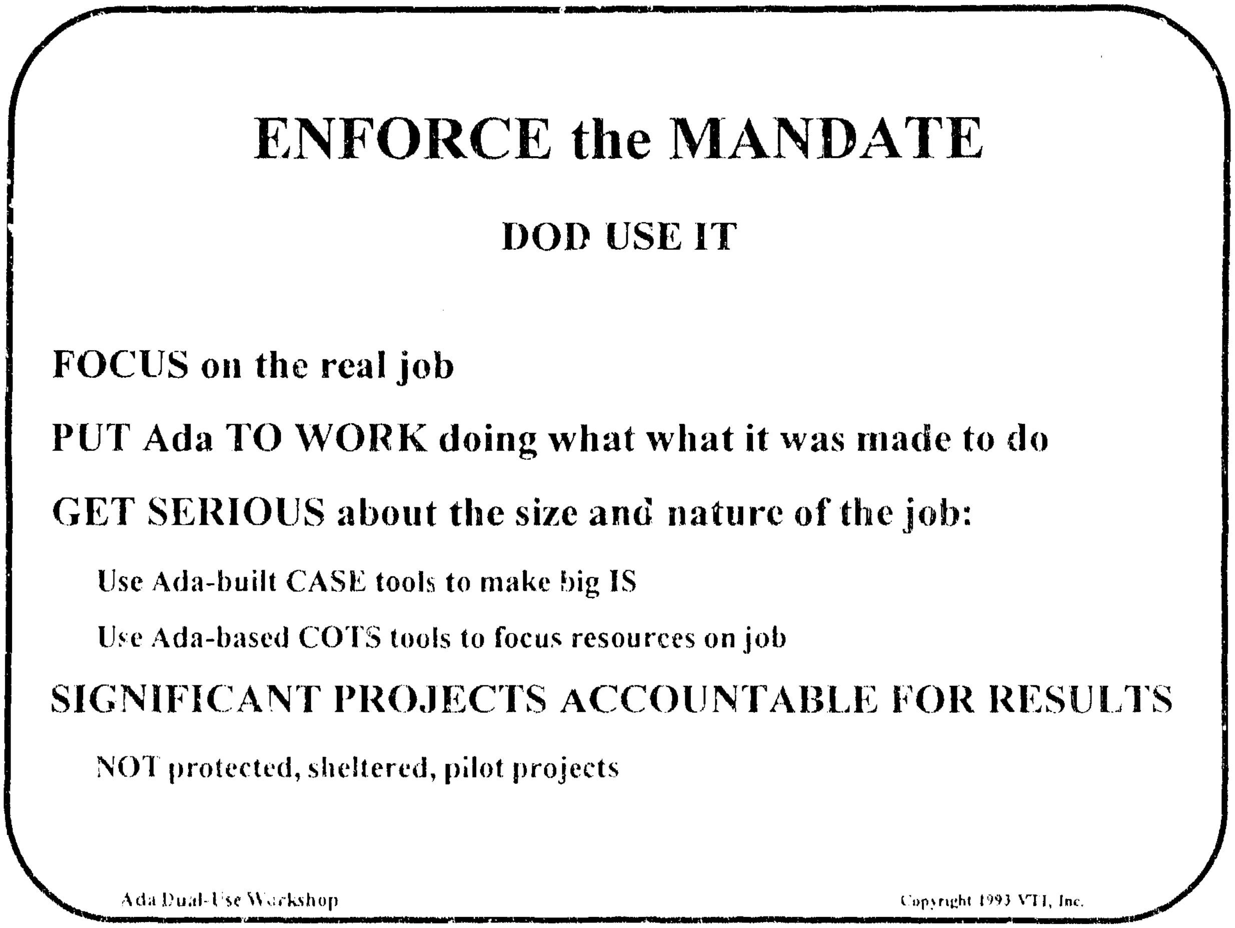
HUGE market exists, and growing

Ada - quantum leap in COMPETITIVE edge

refine Ada usage on solving big IS

develop expertise and cadre in solving big IS

gain track record and publicity from DOD



ENFORCE the MANDATE

DOD USE IT

FOCUS on the real job

PUT Ada TO WORK doing what what it was made to do

GET SERIOUS about the size and nature of the job:

- Use Ada-built CASE tools to make big IS
- Use Ada-based COTS tools to focus resources on job

SIGNIFICANT PROJECTS ACCOUNTABLE FOR RESULTS

- NOT protected, sheltered, pilot projects

ENFORCE the MANDATE

Turn on STANFINS to document maintenance benefits

Put Ada to work on real projects

- produce big IS success stories
- increase cadre
- increase expertise of solving big IS problems

Document & publicize success stories ala GAO reports

- dispel Ada myths with the facts
- remove stigma of Ada for intellectual/embedded use only

Make Ada-built tools easy to use

- put on contract vehices
- get procurement points for Ada-built tools

Participant Briefing
Richard Banta
Westinghouse Electric Corporation

(Presentation materials were submitted in handwritten form. A typed copy was created by the Ada Dual-Use Workshop staff.)

Ada Dual-Use Workshop

R. C. Banta
Westinghouse Electric Corp.

October 19, 1993

Question 1

What can we do to increase the market for Ada goods and services within the non-DoD sectors?

- Advertise to colleges and universities so that they become aware of the availability of low cost Ada compilers for PCs.
- Encourage universities to teach Ada courses.
- Encourage legislation of Ada usage for the entire Federal Government as the language of choice.

Question 1 (cont.)

- Step up the advertising of the low life cycle cost of software when Ada is used.
- Advertise the reuse advantages of Ada for commercial families of software.

Question 2

What investments should we make and iniatives should we take to increase the commercial use of Ada outside of defense applications?

- Encourage legislation of Ada usage for the entire Federal Government as the language of choice.

Question 2 (cont)

- Invest in expanding Ada Clearinghouse to distribute free and low cost compilers to universities.
- Expand the Ada Clearinghouse charter to make it the technical information source for Ada technical data and to be the Q&A focal point for commercial users.

Question 3

What can we do to increase Ada's appeal, adoption and popularity?

- Make better known Ada success stories especially concerning the cost savings achieved by using Ada.
- Have Ada Clearinghouse membership in standards groups such as IEEE and ANSI.

Question 3 (cont)

- Offer a scholarship/financial reward to the college student who writes the best computer game using Ada.

Participant Briefing
Bonnie Dancy
EVB Software Engineering

EVB Software Engineering, Inc.

5303 Spectrum Drive
Frederick, Maryland 21701
Tel: (301) 695-6960
FAX: (301) 695-7734
email: info@evb.com

Presented by: Bonnie Dancy, President

A Software Technology Transfer Firm

❑ Exclusively Ada since 1984

- ○ Developed curriculum to teach Software Engineering in Ada using an object oriented paradigm
- ○ Provided to government and commercial "early adaptors"

❑ In 1985 developed proprietary products for the Ada software development market.

The Initial Excitement

- Technical expectation.
  - Ada viewed as a state-of-the-art language.
  - A standard language to end duplication and waste.
- Revenue expectation: a Guaranteed Market.
  - Mandate would ensure billion dollar market.
  - High Technology, February, 1983.
    "Ada is going to have a dramatic economic impact," - Jean Ichbia, Alsys.

A "Flagship" Reuse Product

- GRACE™, (Generic Reusable Ada Components for Engineering).
  - An early COTS product, developed internally, funded from cash flow.
  - A library of data structures licensed in source code to end users.
  - Tested, documented, warranted - and supported!
  - Customers included Nokia, NTT, and other early adaptors of Ada for engineered software.

Dual-Use Workshop, October 1993

What really happened.

- ❐ User Resistance:
  - ○ A culture shift meant extensive training.
  - ○ Ada tools were expensive!
  - ○ No incentives to use COTS, or to pursue reuse.
- ❐ Ada Mandate unsuccessful – lost confidence in projected market:
  - ○ Small market -> Low volume -> High prices, etc.
  - ○ Low return on investment.

EVB

Dual-Use Workshop, October 1993

Our Current Strategy

- ❐ Build engineered products, high reuse and low support costs.
- ❐ Joint ventures for Ada products
  - ○ Strategies: Ada graphics products
  - ○ GRAMMI from ESL, Inc.
  - ○ Bindings for Itasca (OODBMS)
- ❐ Commercial applications.

EVB

Dual-Use Workshop, October 1993

Current Efforts

- □ Examples of Ada products for the Commercial Market: Reuse & Portability
- □ Cadwin: a next generation CAD tool
- □ Heragraph for the … market
- □ Redefining service.

Dual-Use Workshop, October 1993

#1. How to increase the market for Ada goods and services within non-defense sectors?

- □ Promote Ada to other government sectors
- □ Federal funding of state-to-state technology transfer to promote reuse.
  - ○ State systems run $40-$100 million; low percentage of reuse
  - ○ Example: 1980 - 1990; $950 million spent on Human Resource Programs

Dual-Use Workshop, October 1993

# 2. What investments can we make? Initiatives should we take?

- □ Enforce the mandate – now!
- □ Support the Ada commitment.
- □ Develop business strategies which promote reuse.
- □ Encourage Research & Development; provide Tax credits for investment.
- □ Support the SBIR program.

Dual-Use Workshop, October 1993

# 3. What to do to increase Ada's appeal, adoption and popularity?

- □ Enforce the mandate.
- □ Promote the successes of Ada.
- □ Provide financial incentives for reuse.
- □ Promote technology reinvestment.
- □ Buy commercial Ada products! Walk your talk!

Participant Briefing
James Thomes
Aetech, Incorporated

VENDOR STAKEHOLDER

SHRINK WRAP

James T. Thomes
Colonel USAF(Ret)
President, AETECH Inc.

AETECH
Ada Software Development & Tools

AETECH, Inc.

"Ada Environment Technologies"

- *Ada Software Development and Tools*
- *Founded in 1984*
- *Past Commander of Air Force Computer Training*
- *Established Air Force Ada Training Program*
- *Ada Contractor – STARS, Ada9X, CARDS, KWI, CINCOM*
- *Ada Vendor – 30+ Products written in Ada for Ada*
- *Standard Ada Environments for Desktop III and Desktop IV*

Recommendations:

To increase the use of Ada within commercial technologies, the government should:

1. Make a Firm DoD Commitment
2. Develop and Maintain Standards...not Software
3. Popularize Ada

Recommendation:

"Who will answer the call of the uncertain trumpet?"

1. **Make a Firm DoD Commitment**
2. Develop and Maintain Standards...not Software
3. Popularize Ada

A Simple Business Case

- *Does a need exist for the product?*
- *Can we build a new or better product?*
- *Can we compete at a profit?*
- *Should we risk the capital?*

Does a Need Exist for Ada Products?

Ada must be used!

The Department of Defense mandated Ada in 1983. (DoD 5000.3)

Public Law mandated Ada in 1991 (PL 101-511).

Ada should be used by industry and the government!

Ideal for large complex systems.

Improves quality and reliability

Reduces life cycle costs

Internationally accepted

Ada should be used in academia!

Embodies principles of Software Engineering

Allows instruction in concurrency, strong typing, exceptions...

Ada should be cost-effective for vendors!

Portability increases the market for vendors products.

Reusability reduces R&D , reliability reduces bugs, maintainability makes it easy to fix them.

Does a Need Exist for Ada Products?

Market Perceptions say No!

Ada is far more reliable and almost as efficient as C but...
... 'C' is still used for most DoD applications

Ada tools and libraries now exist for MIS systems but...
... obsolete COBOL systems are still being maintained

Ada educational products are priced at Pascal and C but...
... No Service Academies use Ada for Computer Science I

Ada has been mandated by the Department of Defense, but...
... the mandate is largely ignored!

The Business Case

Does a need exist for Ada?	➡	??
Can we build new or better products?	➡	Yes
Can we make a profit?	➡	??
Should we risk the capital?	➡	No

The Certain Trumpet

- *Enforce Existing Ada Mandates*
- *Use Ada for CS-1 at Service Academies*
- *Achieve Unanimity at this Meeting*
- *Publicize DISA Commitment*

DoD Regulation 3000.3

July 1983

The intent of the law is that you use Ada:

"The Ada programming language shall become the single standard computer programming language for all defense mission critical applications."

Public Law 101-511

November 1990

Unless you can prove otherwise, you must use Ada, because:

> *"Not withstanding any other provision of law, after June 1, 1991, where cost effective, all Department of Defense software shall be written in the programming language Ada, in the absence of special exemption by an official designated by the secretary of Defense"*

The Certain Trumpet

Enforce Existing Ada Mandates

Use Ada for CS-1 at Service Academies

Achieve Unanimity at this Meeting

Publicize DISA Commitment

Service Academies CS-1

"Setting the Example in 1993"

USAFA	—	Pascal
USMA	—	Pascal
USNA	—	C (Now adopting C++)

The Certain Trumpet

- *Enforce Existing Ada Mandates*
- *Use Ada for CS-1 at Service Academies*
- *Achieve Unanimity at this Meeting*
- *Publicize DISA Commitment*

VENDOR STAKEHOLDER

PERSONAL COMPUTERS

James T. Thomes
Colonel USAF(Ret)
President, AETECH Inc.

AETECH
Ada Software Development & Tools

AETECH, Inc.

"Ada Environment Technologies"

- *Ada Software Development and Tools*
- *Founded in 1984*
- *Past Commander of Air Force Computer Training*
- *Established Air Force Ada Training Program*
- *Ada Contractor – STARS, Ada9X, CARDS, KWI, CINCOM*
- *Ada Vendor – 30+ Products written in Ada for Ada*
- *Standard Ada Environments for Desktop III and Desktop IV*

Recommendations:

To increase the use of Ada within commercial technologies, the government should:

1. Make a Firm DoD Commitment
2. Develop and Maintain Standards...not Software
3. Popularize Ada

Recommendations:

To increase the use of Ada within commercial technologies, the government should:

1. Make a Firm DoD Commitment
2. Develop and Maintain Standards...not Software
3. **Popularize Ada**

Supply and Demand

If few people are using Ada then

- *Ada products will be expensive*
- *Reusable Ada software will be rare*
- *DoD Ada systems will be costly and risky*

If many people are using Ada then

- *Ada products will be competitive*
- *Ada quality will be increased*
- *DoD systems will be cheaper and better*

Therefore, it makes sense for the government to help popularize Ada!

Thoughts on Popularizing Ada

"Ada must become the language of choice...
not the language of mandate!"

"People must use Ada because they want to..
not because they have to!"

"Ada must come from the bottom up, not the top down!"

"If Ada is ever to be popular, then Ada compilers
for the PC must have all the features of other
languages, and must be fun and easy to use!"

PC Compilers (1984)

	Basic	Turbo-Pascal	Ada	*IntegrAda*
Validated Compiler			✓	✓
Screen Interfaces	✓	✓		✓
Integrated Environment	✓	✓		✓
640K Memory	✓	✓		✓
No Run-time Royalties	✓	✓		✓

PC Compilers (1994)

	Visual C++	Standard Ada	*ImegrAda* for Windows	Ada9X
Validated Compiler		✓	✓	✓
Common Libraries	✓		✓	
Object Oriented	✓			✓
Language Pragmas		✓	✓	
GUI Environment	✓		✓	
GUI Libraries	✓		✓	
GUI Design Tools	✓		✓	

State of the Art of Ada

"You can now do anything in Ada that you can do in C or C++ and do it faster, better, and more reliably; therefore, there is no reason to grant waivers for the use of the C programming language!"

Recommendations:

To stimulate the use of Ada, the government can help popularize Ada by:

Refusing to grant waivers

PC applications of any kind (C++).

MIS upgrades (COBOL)

Artificial Intelligence (LISP)

Windows (C or Visual Basic)

DBMS (Oracle,Informix,etc)

Demanding the State-of-the-Art in DoD Systems

Modernize Ada PC's in DoD Computer Schools

Mandate Ada in Service Academies

Specify State-of-the-Art capabilities in new RFP's

Mandate Technology Upgrades on existing PC contracts (DTIV)

VENDOR STAKEHOLDER

LANGUAGE TOOLS

James T. Thomes
Colonel USAF(Ret)
President, AETECH Inc.

AETECH
Ada Software Development & Tools

AETECH, Inc.

"Ada Environment Technologies"

- *Ada Software Development and Tools*
- *Founded in 1984*
- *Past Commander of Air Force Computer Training*
- *Established Air Force Ada Training Program*
- *Ada Contractor – STARS, Ada9X, CARDS, KWI, CINCOM*
- *Ada Vendor – 30+ Products written in Ada for Ada*
- *Standard Ada Environments for Desktop III and Desktop IV*

Recommendations:

To increase the use of Ada within commercial technologies, the government should:

1. Make a Firm DoD Commitment
2. Develop and Maintain Standards...not Software
3. Popularize Ada

Recommendations:

To increase the use of Ada within commercial technologies, the government should:

1. Make a Firm DoD Commitment
2. **Develop and Maintain Standards...not Software**
3. Popularize Ada

Counter-Incentives to Commercial Ada

"How can I compete against the federal government?"

GNU-Ada

Free Ada compiler
Licensed to "Free Software Foundation, Inc."
Unvalidated
Funded by Ada9X at $1 Million +
Advertised in government Ada journals

AdaSAGE

Free Library of DBMS and User Interfaces
High Quality, Well Engineered
Developed and Maintained by DOE/INEL
Funded by DoD
In Widespread Use throughout DoD

The Business Case

- *Does a need exist for the product?*
- *Can we build a new or better product?*
- *Can we make a profit?*
- *Should we take the risk?*

The Business Case

Does a need exist for the product? ➠ Yes

Ada embodies principles of software engineering

Ada has features not available in other ıanguages

Can we build a new or better product? ➠ Yes

AETECH's "LearnAda"

Integrated, easy to use, PC-based system

Includes compiler, editor, examples, CBT

Air Force Ada Training Program Curriculum

Priced at $79.95 in Bookstores

Comparable to Pascal and C

Can we compete at a profit? ➠ No

GNU-Ada compiler is free

Advertised in government Ada journals

Reviewed by government advocates

Distributed by government agencies

The Business Case

DBMS and User Interface Libraries (AdaSAGE)

Does a need exist for the products? ➠ Yes

Ada is required for MIS applications.

MIS applications require DBMS and User Interfaces

Ada has no DBMS and User Interfaces

Can we build a new or better product? ➠ Yes

AdaSoft AdaManager

AdaSoft Textual User Interfaces

AdaSoft Graphical User Interfaces

AETECH Ada User

AETECH Ada Graphics

Can we compete at a profit? ➠ No

Government funds AdaSAGE

Distributes free

Recommendation:

To increase the commercial use of Ada

The Government cannot:

Compete with commercial firms

Perform "Work for Others" which competes with commercial firms

The Government should not:

Develop Ada compilers

Develop Ada tools and libraries which already exist.

Require the use of specific products.

Promote and advertise government software.

The Government should:

Establish Configuration Control of Ada applications

Develop Standards for Ada Libraries and Tools

Establish Validation Processes and Agencies for Standards

Test and Certify Vendor Implementations

An AdaSAGE Example

- *Ada Libraries*
- *Developed by the Government*
- *Maintained by the Government*
- *Distributed Free by the Government*
- *Reduces the cost of software development*
- *Discourages Commercialization*
- *Limits State-of-the-Art*

AdaSAGE Recommendations

"Develop and Maintain Standards...not Software"

- *Continue Funding of Existing AdaSAGE Development*
- *Assure AdaSAGE works with all validated Ada compilers*
- *Do not fund new features that compete with COTS products*
- *Transition INEL from AdaSAGE Development to Validation*
- *Fund the development of an AdaSAGE Test Suite*
- *Fund the establishment of an AdaSAGE Validation Office*
- *Release the AdaSAGE source code to vendors*
- *Subsidize initial commercial implementations and validation*

The Business Case

AdaSAGE Validation Vice Development

Does a need exist for the products? ➠ Yes

Ada is required for MIS applications.

MIS applications require DBMS and User Interfaces

Ada has no DBMS and User Interfaces

AdaSAGE would be approved to support Ada

Can we build a new or better product? ➠ Yes

Faster, smaller, better AdaSAGE implementations

New operating systems

New computers

New Graphical User Interfaces

Can we compete at a profit? ➠ Yes

Government is not a competitor but a supporter

Products do not compete with free software

Participant Briefing
Michael Sapenter
Telos Corporation

SOFTWARE ENGINEERING DIRECTORATE,
FIRE SUPPORT SOFTWARE ENGINEERING

CURRENT FIRE SUPPORT SYSTEM PROFILE

TACTICAL SYSTEMS	TARGET	LANGUAGE	#SLOC	STANDARD
BCS	ATCCS LCU	Ada	85K	DOD-STD-2167A
MLRS FDS	ATCCS TCU/LCU	Ada	104K	DOD-STD-2167A
MLRS FMS	ATCCS LCU	Ada	84K	DOD-STD-2167A
ATACMS-1	ATCCS TCU/LCU	Ada	8K	DOD-STD-2167A
MLRS-CDPU	MLRS FDDM	Ada	128K	DOD-STD-2167A
FIREFINDER CTS	ATCCS LCU	Ada	16K	DOD-STD-2167A
FO-CC	ATCCS SHTU	Ada	35K	DOD-STD-2167A
FO-MSR	ATCCS SHTU	Ada	44K	DOD-STD-2167A
FO-FIST	ATCCS HTU	Ada	21K	DOD-STD-2167A
LAMPSS	ATCCS LCU	Ada	31K	DOD-STD-2167A
SYSTEM TO BE TRANSITIONED				
IFSAS	ATCCS LCU	C	280K	BEST COMMERCIAL PRACTICES
AFATDS	ATTCCS TCU	Ada	550K	DOD-STD-2167A
SUPPORT SYSTEMS				
ABTCS	ATCCS LCU, SHTU	Ada	26K	MIL-STD-483 MIL-STD-490
FS3	ATCCS LCU, SHTU	Ada	23K	MIL-STD-483
SYSTEM TO BE CONVERTED				
FIREFINDER Q-36	ATCCS LCU	Ada	TBD	DOD-STD-2167A
FIREFINDER Q-37	ATCCS LCU	Ada	TBD	DOD-STD-2167A
MDS	ATCCS LCU	Ada	TBD	DOD-STD-2167A

US ARMY
COMMUNICATIONS
ELECTRONICS COMMAND

HG0335

SOFTWARE ENGINEERING DIRECTORATE,
FIRE SUPPORT SOFTWARE ENGINEERING

FIRE SUPPORT SYSTEMS HISTORY

TACTICAL SYSTEMS	TARGET PLATFORMS	LANGUAGES	#SLOC	STANDARDS
TACFIRE BCD	AN/GYK-12	TACPOL/MOL	264K	MIL-STD-483 MIL-STD-490
TACFIRE BN	AN/GYK-12	TACPOL/MOL	294K	MIL-STD-483 MIL-STD-490
L-TACFIRE	AN/PVC-1	C	320K	MIL-STD-2167A
BCS	AN/GYK-29	SIR	356K	MIL-STD-483 MIL-STD-490
MLRS FDS	AN/GYK-29	SIR	272K	MIL-STD-483 MIL-STD-490
FIREFINDER Q-36	AN/TPQ-36	ULTRA 16	250K	BEST COMMERCIAL PRACTICES
FIREFINDER Q-37	AN/TPQ-37	ULTRA 16	250K	BEST COMMERCIAL PRACTICES
FIREFINDER	HP-900/360	FORTRAN, C	55K	DOD-STD-2167A
FIREFINDER FMS (Q-36/Q-37)	AN/TPQ-36, AN/TPQ-37	ULTRA 16	265K	BEST COMMERCIAL PRACTICES
MDS	AN/GYK-19	ROHM ASM, SUPERSIMP ASM, DATA GENERAL ASM, C	113K	BEST COMMERCIAL PRACTICES
TF-DMD	AN/PSG-2	PL/M	11K	MIL-STD-483 MIL-STD-490
FIST-DMD	AN/PSG-5	PL/M	7K	MIL-STD-483 MIL-STD-490
FO-DCT	AN/PSG-2	C	22K	BEST COMMERCIAL PRACTICES
SUPPORT SYSTEMS				
ABTCS	AN/GYK-12	FORTRAN	14K	MIL-STD-483 MIL-STD-490
TSS	AN/GYK-29, AN/GYK-12	FORTRAN	27K	BEST COMMERCIAL PRACTICES

US ARMY
COMMUNICATIONS
ELECTRONICS COMMAND

Participant Briefing

Goran Hemdal

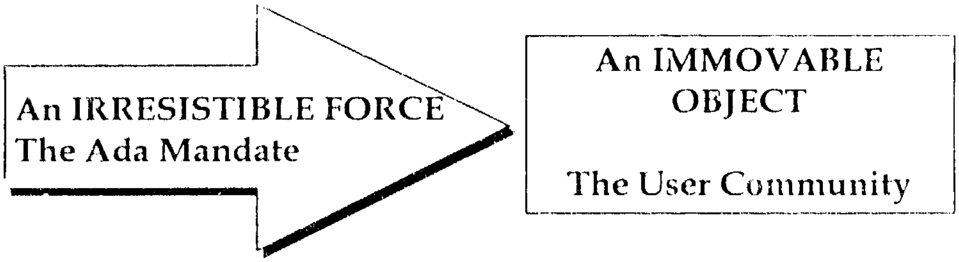
RJO Enterprises, Incorporated

ADA DUAL USE WORKSHOP

Some Points of View

Presented by:
Goran Hemdal
VP Software
RJO Enterprises, Inc
Lanham, Maryland

A Key Problem



An IRRESISTIBLE FORCE
The Ada Mandate

An IMMOVABLE OBJECT

The User Community

The Mandate may not in Itself be Sufficient
to Cause the User Community to Move
(It has not, so far)

What are the Factors that Impede
Movement?

A Success Story

The USNA High Speed Fleet Broadcast System

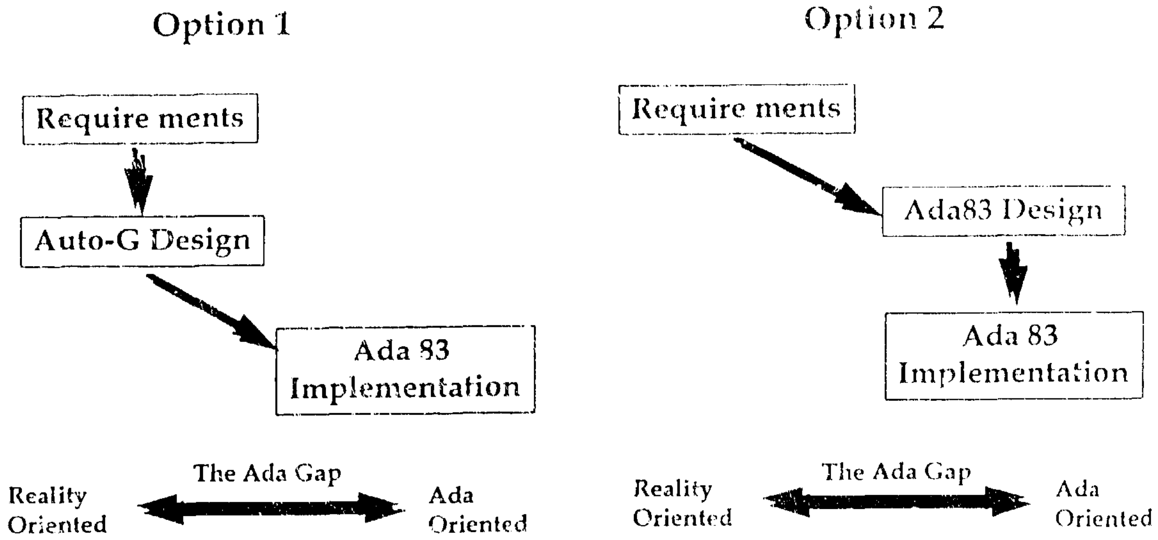
- ★ Real Time Embedded System
- ★ Mandated Software Documentation in Ada
- ★ RJO delivered the HSFB System and Software:
  - On Time
  - Within Budget
  - To Specification (as existing at the time of Delivery)
- ★ System has been Successfully Installed aboard Operational Ships and Shore Stations
- ★ Delivery of more than 50 Systems to Date

A Success Story

The USNA High Speed Fleet Broadcast System

- ★ System Control Software Developed and Documented using RJO's Auto-G CASE Tool Set
  - Requirements Analysis
  - Design (All Levels)
- ★ The Delivered Ada Code generated using RJO's Autoada Code Generator
- ★ The Generated Ada Code Compiled and Executed without Manual Modification or Intervention
- ★ Maintenance Performed Only at the Design Level
  - Any Design Modifications Carried Out Using Auto-G
  - New Ada Code is Generated from the Modified Design (No Hacking or Patching)

What Were the Design Options?



Option 1

Require ments

Auto-G Design

Ada 83 Implementation

Reality Oriented

The Ada Gap

Ada Oriented

Option 2

Require ments

Ada83 Design

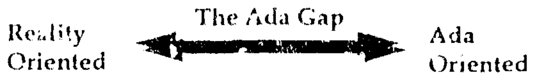
Ada 83 Implementation

Reality Oriented

The Ada Gap

Ada Oriented

The Ada Gap is the Real Problem



Reality Oriented

The Ada Gap

Ada Oriented

- ★ The Ada Gap is Significant, in Particular for Real Time Embedded Systems
- ★ For the HSFB System the Ada Gap was Successfully Bridged by the automated Ada Code Generation Anchored in Auto-G
- ★ If the Ada Gap is Too Wide and Cannot be Successfully Bridged the User Community Will Not Jump
- ★ Mandated Use of Ada will not Reduce the Ada Gap

How Can the Ada Gap Be Bridged or Reduced?

★ By Using Auto-G or some CASE Tool with Similar Capabilities to Generate Ada83 Code

- Does Not Reduce the Ada Gap
- Will Not Permanently Bridge the Ada Gap Unless the Generated Code is 100% Executable (No Hacking or Patching of the Generated Code
- Significant Performance Overhead

★ By Using Ada9X

- Significantly Reduces the Ada Gap.
- The Remaining Gap may be Bridged much more Simply than for Ada83 and Without the Performance Overhead

What Can We Do to Increase the Market for Ada Goods and Services Within the Non-Defense Sectors?

★ Increase Ada's Appeal, Adoption and Popularity

What Can We Do to Increase Ada's Appeal, Adoption and Popularity?

★ Anything that Bridges or Reduces the Ada Gap, Including but not Limited to Accelerated Introduction of Ada9X

What Investments Should We Make and Initiatives Should We Take to Increase The Commercial Use of Ada Outside Defense Applications?

★ Any that Help Bridge or Reduce the Ada Gap, Including but not Limited to Accelerated Introduction of Ada9X

What Can We Do to Increase the Market for Ada Goods and Services Within the Non-Defense Sectors?

- ★ Increase Ada's Appeal, Adoption and Popularity

What Can We Do to Increase Ada's Appeal, Adoption and Popularity?

- ★ Anything that Bridges or Reduces the Ada Gap, Including but not Limited to Accelerated Introduction of Ada9X

What Investments Should We Make and Initiatives Should We Take to Increase The Commercial Use of Ada Outside Defense Applications?

- ★ Any that Help Bridge or Reduce the Ada Gap, Including but not Limited to Accelerated Introduction of Ada9X